

**Visualization of Rail Passenger Survey Data
and its Delivery as a Web Service**

Abdulaziz Albatli
MSc Computing and Management
2011/2012

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of student) _____

Summary

Data visualisation is a trending topic in the IT community as it is clearly necessary to provide evidence of how data differs in patterns and to show changes in trends. This project aims to build a web-based system for Tracsis Plc that can provide visualisation analysis of data stored on its database through uploading the company's survey reports. These reports produced as a result of surveys are not a directly informative representation of the data gathered, which is generally about the number of passengers getting on/off trains and in/out of train stations. Besides, the information is difficult and sometimes complex to be easily and readily assimilated by their clients. Thus, the system is thought to add a meaningful context to the data by generating a number of desired visual forms, that is, Line Charts, based on the configuration specified by the user. Also, the system offers a lot of functionalities to the user in order to allow more control over the system to be achieved.

Acknowledgment

I would first like to thank both my supervisor Dr. Raymond Kwan and my assessor Dr. Marc de Kamps for their invaluable and patient assistance and supervision, natural kindness, and continuous encouragement. Also, Mr. Darran Moss, representative of Tracsis Plc, has been really helpful which I really appreciate.

I would like to also thank my friends and colleagues who gave me such courage and support that I needed. I especially thank Khalid Alhammad and Ibrahim Alzamil for their thoughtful suggestions and support.

Lastly and most importantly, I thank my wife and family for their love, understanding, and patience during the stressful times I have gone through. I take this moment to thank Ramzi and Faris for their unforgettable advice, during my lifetime abroad, that has changed my life for the better. Without you all, my success would not have been possible.

Glossary of Terms

Term	Meaning
A, Alight or Alighting;	Number of passengers leaving a train.
J, Join or Joining;	Number of passengers joining a train.
OTA	Total number of passengers on train on arrival.
OTD	Total number of passengers on train on departure.
WBS	Work Breakdown Structure
OCR	Optical Character Recognition

List of Figures

Figure	Title
Figure 1.1	Initial System Design
Figure 1.2	Proposed Iterative Approach
Figure 3.1	A Snapshot On Train File
Figure 3.2	A Snapshot Calling Point File
Figure 3.3	A Snapshot of Terminus File
Figure 3.4	A Screenshot of the Final Version of The System
Figure 3.5	Steps to Generate Visual Form(s)
Figure 3.6	The Final Database Design
Figure 3.7	Insertion Code for On Train Data
Figure 3.8	Insertion Procedure for On Train Data
Figure 3.9	System Areas
Figure 3.10	Login Code
Figure 3.11	Project Registration Page
Figure 3.12	On Train Option 1 Analysis
Figure 3.13	On Train Option 2 Analysis
Figure 3.14	Terminus Option 1 Analysis
Figure 3.15	Terminus Option 2 Analysis
Figure 3.16	Calling Point Option 1 Analysis
Figure 3.17	Calling Point Option 2 Analysis
Figure 4.1	On Train Option 1 Result
Figure 4.2	On Train Option 2 Result
Figure 4.3	Terminus Option 1 Result
Figure 4.4	Terminus Option 2 Result
Figure 4.5	Calling Point Option1 Result
Figure 4.6	Calling Point Option 2 Result, Single
Figure 4.7	Calling Point Option 2 Result, All
Figure 5.1	Debugging Report

List of Tables

Table	Title
Table 5.1	Manual Testing
Table 5.2	User Acceptance Test

Content

Summary	ii
Acknowledgment	iii
Glossary of Terms	iv
List of Figures	v
List of Tables	v
Table of Content	vi
1. Introduction	1
1.1. Overview	1
1.2. Aim and Objectives	1
1.3. Minimum Requirements	2
1.4. Further Enhancements	2
1.5. Deliverables	3
1.6. Relevance	3
1.7. Methodology	3
1.7.1. Software Development	3
1.8. Project Plan	6
1.8.1. Key Activities	6
1.8.2. Milestones	6
2. Review of Literature and Technologies	7
2.1. How Visualization Works	7
2.1.1. Visualisation Systems	9
2.1.2. Visual Forms	9
2.2. Data Collection	10
2.3. Data Mining	11
2.3.1. Types of Data	12
2.4. Web-Based Systems	12
2.4.1. Definition	12
2.4.2. Advantages	12
2.4.3. Drawbacks	13
2.5. Review of Programming Languages	13
2.5.1. ASP.NET	13
2.5.2. PHP	14
2.5.3. C#	14
2.5.4. HTML & CSS	14
3. Implementation and Project Management	15
3.1. Problem Summary and Proposed Solution	15
3.2. System Functionality	18
3.2.1. Core Features	18
3.2.2. Additional Features	19

3.3. Technology	20
3.3.1. Programming Languages	20
3.4. Development Sequence	21
3.4.1. Database and logic design	22
3.4.2. Admin and Client Areas with Distinguished Roles	24
3.4.3. Data Manipulation	26
3.4.3.1. File Validation and Data Insertion	26
3.4.3.2. Visualising the Data	27
3.5. Project Management	29
4. Results	31
4.1. Data Comparison Before and After Analysis	31
4.1.1. On Train	31
4.1.2. Terminus	32
4.1.3. Calling Point	33
5. Evaluation	35
5.1. Methodology Evaluation	35
5.1.1. Pros	35
5.1.2. Cons	36
5.2. System Evaluation	36
5.2.1. Code Testing	36
5.2.1.1. Manual Testing	37
5.2.2. User Acceptance Test	37
5.2.3. Survey of Client	38
5.3. Results Evaluation	39
5.3.1. Chosen Visual Form	39
6. Conclusion and Future Work	40
6.1. Outline of the Final Product	40
6.1.1. Minimum Requirements	40
6.1.2. Further Enhancements	41
6.2. Limitations	42
6.3. Future Work and Developments	42
7. Bibliography	44
8. Appendices	49
8.1. Appendix A (Reflection)	49
8.2. Appendix B (Interim Report Feedback)	51
8.3. Appendix C (Requirements Catalogue)	53
8.4. Appendix D (WBS)	55
8.5. Appendix E (Project Management Documents)	59
8.6. Appendix F (User Guide)	76
8.7. Appendix G (Manual Testing)	92
8.8. Appendix H (Evaluation Survey Forms)	96

1. Introduction

1.1. Overview

Tracsis Plc, the client, is a developer and a provider of data capture, and reports technologies to the transport industry [1]. A company related to Tracsis Plc is Passenger Counts, which specialises in rail market research and passenger counting [2]. Based on the request of rail companies or the Department for Transport (DoT), they carry out a survey of the number of passengers for a specific line, season, or rail station. For example, for a train departing from London King's Cross to Leeds, they count the total number of passengers joining the train from the origin station and those who join or leave the train at the in between stations, along with those leaving the train at the destination station. As a result of a complete survey, a report is produced.

The report produced as a result of the survey is not a directly informative representation of the data gathered. Besides, it is difficult and sometimes complex to be easily and readily assimilated by the client. Thus, a visualisation of the data delivered as a web-based system has been proposed as a solution. An explanation of both the problem and the proposed solution will be presented later in the report.

1.2. Aim and Objectives

The overall aim of this project is to propose and implement a web-based system that provides visualisation analysis of raw data, gathered by Tracsis Plc, and fed into it. The raw data is generally about the number of passengers getting on/off trains and in/out of train stations. Specifically, this project aims to provide a prototype of some aspects of data visualisation, such as charts, graphs and tables. In addition, the main users of the system would be the clients of Tracsis Plc, in other words, rail companies and the Department of Transport.

In order to achieve the proposed aim, a number of objectives have been outlined:

- Understand and discuss where, when, how and why data is collected more thoroughly in order to have a better insight when analysing the requirements.
- Understand and discuss web-based systems, including how they are built.
- Understand how to implement visualisation of data in different ways that are relevant to the type of data being analysed.

1.3. Minimum Requirements

The following minimum requirements have been identified after meeting the client:

- Identify and implement a method for visualising the data from files fed into the system, to create meaningful information.
 - Identify the data that is required out of other non-useful data in the files.
 - Convert the data into a database-format and store it on the database.
 - Identify what needs to be understood from the visualised data.
- Identify and implement a suitable way of displaying the graphical forms (Graphs, Charts, or Tables) on the web application.
- Implement the system to be friendly (ease of use) and adaptable/configurable to the three different styles of files in terms of their layout.
- Identify and implement an appropriate evaluation method for the system.

1.4. Further Enhancements

The following enhancements are suggested as ideas to be achieved when there is enough time and resources:

- Identify and implement a method for comparing between files or projects (seasonal data comparison).
- Implement more functionality in terms of managing the system.
 - Retrievable archived projects;
 - Update previously created projects;
 - Projects details
 - Replace relevant file(s)
 - Printable Visual Forms;
 - Chart
 - Title
 - Comment
 - Print date
- Make the system runnable on local machines in addition to being online so that the administrator will not be bound to internet connectivity when managing the system.

1.5. Deliverables

The main deliverable is a web-based visualisation system (product) that satisfies the minimum requirements, and an evaluation of its dependability.

1.6. Relevance

This project has been chosen due to the fact that it is challenging as it requires more skills and knowledge in order for its aim and requirements to be accomplished. Besides, it builds on the knowledge and skills gained from a number of modules studied on the course MSc Computing and Management. Marketing for Management and Understanding Organisations were useful in terms of developing the skills relating to research, critique, and management which are needed to build this project. Also, Problem Solving with Computers and Advanced Distributed Systems ensured familiarisation with the basics of programming and Web Services, and presented the skill of logic structuring of coding required in the implementation phase. Besides, Business Information Systems was particularly useful due to the inclusion of crucial skills for adopting an appropriate methodology for this project. Lastly, Techniques for Knowledge Management introduced key issues in knowledge and data management, information visualisation, and databases, which are all important for this project.

1.7. Methodology

As stated above, the aim of this project is to propose and implement a web-based system that provides a visualisation analysis of the raw data produced by Tracsis Plc and fed into it. In order to achieve that, adopting a suitable software development methodology is necessary. The purpose of the proposed adoption is to reduce the chances of project failure, which is highly likely in software development [3].

1.7.1. Software Development

After extensive research, the waterfall model has been chosen due to being well-integrated and suitable for developing this project. Actually, choosing this model specifically is due to its clarity and fulfilment of the system development life cycle's stages (SDLC). Also, it fits with the timeline given to produce the system as it is a fast approach [4]. The SDLC's stages are: system analysis, system

design, system implementation, system testing, and system maintenance [5]. All of which will be examined respectively in terms of the proposed project.

First of all, one of the important stages in the SDLC is system analysis because it helps in constructing the right strategy for implementing the system. In fact, the goal for this stage is to identify the requirements that allow the project to be modelled logically [5]. For this project, the requirements have been documented, which is the deliverable of this stage, in a meeting with the client (please see Appendix B for the original requirements catalogue and the updated one).

Second, in the system design phase, a fulfilling physical design will be undertaken once the requirements have been identified and documented. The design is thought to outline the essential inputs and outputs of the system in addition to the user interface. The result of this stage is the system design specification [4, 5]. In terms of the proposed project, an initial design of the web system has been suggested and agreed upon by the client, as shown in the Figure below.

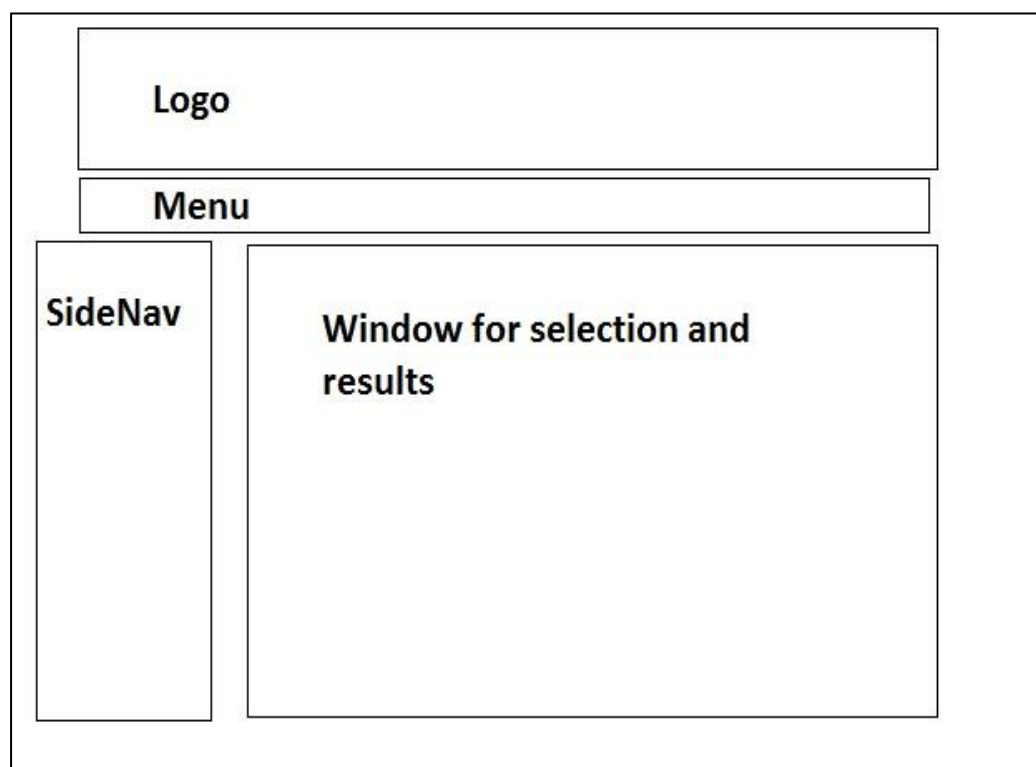


Figure 1.1: Initial System Design

Thirdly, constructing (coding) of the system takes place in the implementation phase. Additionally, delivering a complete functioning and well documented system is the objective of this phase. For this project, a review of a number of programming languages will be conducted later in

this report. The results of this review will be a selection of suitable languages in order to construct the proposed system.

Fourthly, the whole system should be tested before handing it to the client. In fact, two different tests will be used. In terms of the code, the environment that would be used, Visual Studio 2010, automatically debugs before every system run. In case the code becomes complex and long, UNIT testing would be considered to be used so that all of the code will be checked, looking for non-used pathways or any bugs [4]. Also, a number of small tests would be completed by the developer, such as checking buttons and links (manual testing). Additionally, in terms of the requirements set by the client, a user acceptance test would be used. In this test, all the requirements will be reviewed in order to ensure that the finished system fulfils the client's agreed expectations.

Finally, once the client has received the final system, the maintenance phase begins. Noticeably, changes in the environment, in terms of an industry, can happen any time. Sometimes, the client places further requests to be added to the system. Thus, changes to adapt to the new environment or updates, whether they are from the provider or the client, could be undertaken [4, 6]. For example, in terms of this project, the client may want to extend the web system to satisfy other requirements than those in the initial agreement.

In addition, three phases of the water fall model will follow an iterative approach, which are Design, Implementation and Testing. The reasons behind that decision are to minimise the chances of project failure and to ensure achieving the maximum quality given the limited timescale. Figure 1.2 shows the suggested iterative approach.

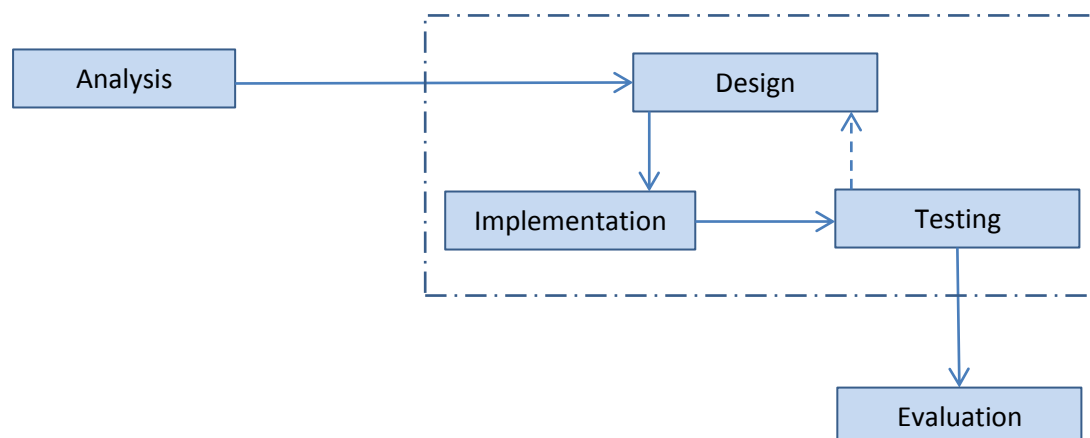


Figure 1.2: Proposed Iterative Approach

1.8. Project Plan

In terms of project management, a Work Breakdown Structure (WBS) and a Gantt chart have been produced. The given timescale has been taken into consideration when constructing the WBS (please see Appendix D). It is proposed that this detailed WBS would help the developer to keep track of each task or phase and not go over budget in terms of time [7]. Besides, the likelihood of the project plan being changed over time due to many reasons, such as illness or work overload, has also been taken into consideration as well.

1.8.1. Key Activities

Initiating: this phase involves meeting with both the supervisor and the client to set up the initial requirements.

Planning: this phase covers constructing the documents needed in terms of project management.

Executing: this phase covers three crucial aspects of the project: a literature review which consists of background research of relevant solutions; analysis which involves understanding the problem more deeply; design which consists of designing the web pages and the structure of the code, and finally, implementation which covers the building of the proposed system.

Evaluation: this phase covers the testing of the system and evaluating its dependability.

Controlling: this phase covers the weekly progress reports of the implementation phase.

Closing: covers the writing up of the dissertation and handing in the product.

1.8.2. Milestones

In order to motivate myself to utilise my knowledge and skills in achieving the aim and the requirements of this project, a number of self-goals (milestones) have been created to act as deadlines or the start of a new phase [8]; these are:

- Decide on Project (28/Feb/2012)
- Start Planning (9/Mar/2012)
- Start Analysis (3/Apr/2012)
- Start Implementation (11/Jun/2012)
- Start Monitoring (8/Jun/2012)
- Start Evaluation (9/Jul/2012)
- Start Writing Dissertation (16/Jul/2012)
- End of Project (30/Aug/2012)

2. Review of Literature and Technologies

Visualisation is an extensive field that covers every discipline ranging from imaging to single theory, and from statistics to computer graphics [9]. There are two branches of visualisation- Information Visualisation, which is about the physical data, and Scientific Visualisation, which concerns the abstract data [10]. The basic general principle of visualisation is to try to extract meaningful information (knowledge) out of abstract data through visual representation by using images, graphics or sound, for example. There are many definitions of visualisation in the literature. The most compelling is by Williams et al: “Visualization is a cognitive process performed by humans in forming a mental image of a domain space” [11].

One of the first examples of visualisation was produced by Harry Beck in 1931. His idea was that the tube map should only include data needed by the passenger. Thus, he redesigned the map for the London Underground, simplifying the complexity of the tube system to a number of horizontal, vertical and 45° diagonal lines [12]. The point was that the passenger will depart with the image in his/her mind.

In addition, one of the issues of visualisation is the nature of data being constantly changed, which results in new information being added or cleansing existing information. In addition, lots of valuable information is not used due to the fact that available data is not being visualised to its greatest potential [13].

Further, there are many visualisation tools ranging from open source to bespoke ones. Also, accessing the final results could be made available either in local machines or via the web, which would enable widespread access to information. In this report, a number of technologies used in building web-based systems will be discussed after investigating visualisation and the tools used. In this chapter, a review of the literature will be presented in the context of this project.

2.1. How Visualisation Works

As has been mentioned, given the difficulty of data, using it to deliver meaningful results requires knowledge from diverse fields, such as: statistics, data mining, graphic design, and information visualisation. In fact, the path from collecting data to producing the final images is rather complicated and consists of a number of operations related to data processing [9].

Scientists have stated that the method of understanding data usually starts with a set of numbers and a question. The recommended steps that form the answer are as follows [13]:

- Acquire: collecting the data, whether from a source over a network or a file on a disk.
- Parse: structuring the data's meaning, and categorising it.
- Filter: eliminating non-relevant data.
- Mine: distinguishing patterns or placing the data in mathematical context by applying methods from data mining or statistics.
- Represent: producing a basic visual model, such as a pie chart.
- Refine: developing the initial representation to make it more visually engaging.
- Interact: manipulating the data or controlling the features to be made visible.

In fact, other scientists have combined the above steps into four logical steps [10]:

- Data Selection: choosing the data to be analysed.
- Data Enrichment: approximating or interpolating raw data - effectively creating a model
- Mapping: transformation of data into a geometric illustration.
- Rendering: allocating visual properties to the geometrical objects (e.g. colour, texture) and generating an image.

From the Tube example mentioned above, it can be understood that information visualisation should start with questions but not from the perspective of the data set. The most important part of understanding data is identifying the question that should be answered; in fact, the more specific the question, the more specific and clear the visual result will be [13]. In addition, even though an exact question could be answered with simple search queries without visual representation, displaying the answer in a visual manner makes it easier and faster to understand [9].

Moving on, visualisation, in general, can be seen in many day-to-day applications [10]. For example, in medicine, 3D pictures of the human anatomy could be visualised using volume rendering which helps in diagnosing diseases more effectively. Also, in climate prediction, just-in-time display of climate data, such as temperature or cloud distribution, is crucial in weather forecasting. In addition, in Computational Fluid Dynamics, the interface between immiscible fluids, that is water and oil, or flow of air around an object, that is, a car, is useful in a variety of studies undertaken nowadays [12].

2.1.1. Visualisation Systems

There are four phases of visualization systems [10]: Subprogram Libraries, Interactive Packages, Visual Programming Systems, and Service-Based Visualisation; the latter two are the most relevant.

In Visual Programming Systems, scientific visualisation is understood as a sequence of simple processing steps, that is, contouring. Additionally, systems provide modules implementing simple stages in a visualisation pipeline where scientists use ‘visual programming’ to connect these modules together [10]. Thus, a number of visualisation tools (systems) have been investigated to assess their reliability for the client in terms of the ability to satisfy their requirements.

First, *Amira* is an object-oriented (OO) visualisation system. It is not based on the data flow model. Instead, data is passed as C++ objects between modules. Also, extending *Amira* is possible by creating new modules [14].

Second, *IBM OpenDX* is also a modular visualisation system based on the dataflow model. The modules can be located on multiple hosts [14].

Third, *IRIS Explorer* is a modular visualisation system based on the data flow model [14]. Extending *IRIS Explorer* is possible by adding code as modules. Collaborative visualisation is supported in this system by modules developed from the COVISA project [14]. This system allows collaborators to share geometric data, raw data or generated images by using different collaborative modules.

Fourth, *AVS* is a system that uses a visual network editor in order to offer an application development environment for visualisation [14]. It can be developed using one of the lower-level facilities, such as GUI toolkit, which results in a more highly tailored interface. *AVS* also relies on an OO model which is reachable via the visual editor allowing access to successively deeper levels of the unit structure [14].

2.1.2. Visual Forms

Tables, charts and graphs are visual representations [15]. To clarify, a *table* includes a set of facts and figures arranged together in columns and rows. A *graph* is a diagram that shows the relationships between two or more sets of numbers or measurements (data), usually as a line or a curve. Finally, a chart is similar to how graph works but, instead, information is represented in different shapes, such as pie or bar charts [15, 18].

The main purpose of using them is to organise information to show relationships and patterns [16]. In fact, scientists and researchers use these visual forms to show their findings or results in order for them to be easily understood and absorbed by readers. Also, using these forms could be suitable for supporting opinions or arguments. Even though these visual forms sometimes only show part of a story, they often lead to better decision making [17]. Thus, it is crucial that a person involved in understanding these visual forms, knows what part of the story they provide. Thus, the goal of these visual forms is to explain the underlying data while enabling the reader to easily identify problematic arrangements that may need further analysis [16].

However, these visual forms are not always a good choice as they may misrepresent the information by misrepresenting relationships between the combined data or perhaps by ignoring some crucial information [15, 16].

2.2. Data Collection

Identifying the question that needs to be answered is a crucial part of understanding the data [13]. It determines the method for gathering the data as well as storing it, in terms of database designs. In addition, researchers state that thinking about how data will be used, and working backward to what was collected, is more important and cleverer than thinking about the data that was collected [13].

For this particular project, the company is commissioned by either a rail company or DoT to carry out a project according to the client [20]. The common task in all projects is counting the number of passengers who use trains and/or train stations. The client has stated that their clients always have questions or objectives they want to answer or achieve. For example, how many people are using Reading station over the Easter holiday? Answering that question will help DoT to carry out work maintenance over the next Easter holiday.

There are three different scenarios implemented in order to carry out those surveys [20]. First, *On Train*, where for a particular train, there are counting staff in every coach. They count the passengers getting in/out the train at every station the train stops at, starting from the origin station, till the destination. For the second scenario, *Calling Point*, the counting-staff are stationed in a particular station and count all passengers who are joining or leaving all trains that pass through that station. Lastly, *Terminus*, whereby the counting-staff are placed in a station and record the number of passengers who are leaving trains that terminate at that station. They also count the number of passengers who are joining trains that originate from that station.

The data gathered can tell many stories. It can identify the peak and non-peak times of the usage of trains or stations. Also, it can provide the total/average number of passengers in a particular time/day/season for a chosen station/train. Further, the benefits of these analyses help rail companies, for example, to add more cars (coaches) for a particular train to cope with demand or vice versa. Also, it could help the DoT to rebuild stations and decide whether to add or remove platforms [20].

Lastly, Tracsis, Plc. uses surveys as a method for collecting data. Also, for those projects to be carried out efficiently and effectively, the company hires ex-rail personnel, conductors, to do the job due to their rich experience [20].

Filtering or mining data, which will be discussed in the next section, is often driven by its original format, which is determined by the way it is gathered and parsed [11].

2.3. Data Mining

Data mining is the process of extracting interesting (implicit, non-trivial, and useful) knowledge from large amounts of data [10]. The data sources can include data warehouses, the web, and so on. In addition, some people consider data mining as a synonym for knowledge discovery from data (KDD), whereas others consider data mining as a step in KDD [21]. Furthermore, according to the literature, the stages of KDD are as follows [21]:

- *Data Cleaning*: removing noise or inconsistent data.
- *Data Integration*: combining data sources, if necessary.
- *Data Selection*: retrieving relevant data from the database.
- *Data Transformation*: transforming data into forms suitable for mining.
- *Data Mining*: extracting data patterns.
- *Pattern Evaluation*: identifying interesting patterns.
- *Knowledge Presentation*: using visualisation to present the mined data.

Data Cleaning is a crucial step in KDD or Data Mining because it involves filling in missing values, correcting inconsistent data, identifying outliers and smoothing out noisy data, and resolving redundancy generated by data integration [22]. A number of reasons why some data might be missing are cited in the literature, such as equipment malfunction; however, there are many methods for dealing with such an instance, such as fitting the missing data automatically with a global constant or the most probable value [21].

2.3.1. Types of Data

Moving on, data mining can be applied to any kind of data only if it is meaningful for a target application [21; 22]. Databases, transactional data, and data warehouses are the most basic forms of data for mining applications. Other forms of data are data streams, networked data, text data, and so on [23]. Those types of data can introduce more challenges, such as mining patterns that carry rich structures and semantics. Also, handling data that carry unusual structures, such as graphs or trees, might be difficult.

2.4. Web-Based Systems

2.4.1. Definition

A web application is a client-server application that is reachable through a network, such as the Internet, and using a web browser as a client [24]. In fact, it is widespread because of the convenience of using a web browser as well as the uninterrupted existence of web browsers. One of the reasons behind the attractiveness is that installing any kind of software on clients' computers for updates or maintenance for the web application is not needed. A number of examples of widely used web applications nowadays are: Wikis, Google Maps, blogs and Flickr [25]. Ruvalcaba gives a simple explanation of web applications, stating that a web application is a 'dynamic website' [26].

Moreover, an arrangement of client-side script (HTML, CSS, etc.) and server-side script (ASP.NET, PHP, MySQL, etc.) is widely used in web applications in terms of the development process [27]. The server-side script deals with all the backend functionalities, such as retrieving and storing the data, while the client-side script deals with the interface or the appearance of the application [27, 28].

2.4.2. Advantages

The key reason that makes web applications popular is because of the advantages and benefits provided [28]. One of these is the fact of not using any resources from the client's side in terms of software installation, which saves CPU processing, for example. In fact, if any updates are needed, they will be done separately by the provider of the web application (service). Secondly, data is secured from any unexpected system crashes. Data would be in a safe place and undamaged in case of the clients' systems going down. Thirdly, location is absolutely not an issue; thus, clients can log onto the web application from anywhere if an Internet connection is available. Lastly, access to the web application would be allowed all day long, so the client is never bound by time or location.

2.4.3. Drawbacks

As it is known, although a web application has advantages, it still has a number of drawbacks. Some of these are: the need for web browser compatibility. If a web browser does not have or support some features, users could be affected by losing some control over the web application. Second, requiring Internet connectivity all the time, which is problematic to fulfil for some users. Thirdly, depending fully on the server, which is dangerous; in case the server goes down for any reason, data could be lost or corrupted and may not be recovered, as has happened to some cloud services such as Amazon's incident in 2011 [29]. Lastly and most importantly, lack of security. Web application level is a desired target for more than 75% of Internet attacks. Therefore, data integrity could be violated on the server even though the communication channel between the server and the client is secure [30].

In order to face those drawbacks, clients should always anticipate the results of such disadvantages. Scientists and experienced people suggest that backups should always take place on a regular basis and stored on different locations, that is, every month [31].

2.5. Languages Used for Creating a Web Application

A number of languages and tools are utilised in order to create the web application, such as ASP.NET, PHP, C#, HTML and CSS, all of which will be explained respectively.

2.5.1. ASP.NET

ASP.NET is known by its flexibility. It is a server-side language used for developing a web application [26]. This means that it runs on a server, although it is better for a web developer to first understand the client-side's technologies, such as HTML. Nevertheless, ASP.NET has some disadvantages. To exemplify, complexity and slow operations are known in ASP.NET, more so than other languages. On the other hand, it has been claimed that one feature that makes ASP.NET stand out from its rivals [32], such as PHP, is the ability to store users' details as they navigate from one page to another, which is sometimes useful for different purposes, such as advertisement tracking. Besides, ASP comes with rich and useful client and server side controls that are used for creating wizards, interactive grids, calendars, and so on [26]. Also, ASP support automatic statefulness which saves the state of controls and some other fields, saving valuable time but leading to performance overheads. One of its major disadvantages is that it does not support automatic unit testing using, for example, NUnit testing tool [28, 32].

2.5.2. PHP

PHP officially stands for “PHP: Hypertext Preprocessor” [33]. It is a server-side programming language and the most widespread programming language on Web Servers. PHP language has evolved through many versions to include or support crucial features such as encryption, session-handling support, and so on. One of its important features is that it uses loosely typed language, which means that it is not required to explicitly create or destroy a variable as it is done internally [34]. Also, its greatest advantage is that it supports 25 different database products, such as Ingres, FilePro and IBM DB2 [33]. On the other hand, its known drawback is its security flaw due to unimplemented or unknown vulnerabilities.

2.5.3. C#

C# has evolved through generations, C# 1.0, 2.0 and 3.0, where each generation adds more features to the language [36]. Even though C# is very simple, with nearly 100 keywords, it is very expressive in developing high modern programming concepts [37]. The keywords that are supported by C# are utilised to develop and implement the three pillars of object oriented programming, which are encapsulation, inheritance, and polymorphism.

2.5.4. HTML & CSS

HTML (Hypertext Markup Language) is the basis of most web pages [35]. Fundamentally, the content of a web page is structured by HTML. In HTML, tags are used to differentiate between paragraphs, tables, headings, forms, and so on.

CSS (cascading style sheets) is all about designing the web page. Rules are used in terms of controlling the appearance or the interface of the web page [38]. These rules, which are written in a separate file, are called by selectors in the HTML file specifying the required elements on which they are to be applied.

3. Implementation and Project Management

This chapter is about the implementation phase of the project, as mentioned in Section 1.7.1, which started after the system analysis and system design phases had taken place. As it has been stated, a full Requirements Catalogue has been documented, Appendix C, along with the updated design sketches which will be explained later. In this chapter, a rich description of the process of building the web system from its early stages to the finished product will be presented. At the end of this chapter, an explanation of how the project was managed will be set out, along with some explanations of some of the management documentations used, including weekly reports and a risk management plan.

3.1. Problem Summary and Proposed Solution

To implement a satisfactory visualisation system, the problem has to be thoroughly and carefully understood [39]. As has been mentioned earlier, at the end of every survey that is done by *Tracsis, plc.*, reports in the format of Microsoft Excel (.xls or .xlsx) are produced. These files can be in three types and each has its own format (layout). Their clients can sometimes find it difficult to extract knowledge out of these files, and if they do, it requires effort and time [20]. Figures 3.1, 3.2 and 3.3 explain how the files look [20].

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1		Station	0620			0635			0650			0705			0720		
2			J	A	OTD	J	A	OTD	J	A	OTD	J	A	OTD	J	A	OTD
3	1	Gospel Oak	16	0	16	12	0	12	33	0	33	28	0	28	0	0	0
4	2	Upper Holloway	7	2	21	12	1	23	7	3	37	6	6	28	0	0	0
5	3	Crouch Hill	5	1	25	3	1	25	9	4	42	12	3	37	0	0	0
6	4	Harringay Green Lane	5	2	28	9	1	33	21	7	56	22	4	55	0	0	0
7	5	South Tottenham	18	0	46	14	4	43	13	5	64	28	2	81	0	0	0
8	6	Blackhorse Road	15	8	53	33	4	72	24	10	78	23	13	91	0	0	0
9	7	Walthamstow Queen's Road	17	0	70	6	3	75	9	3	84	21	5	107	0	0	0
10	8	Leyton Midland Road	16	9	77	9	6	78	10	8	86	17	10	114	0	0	0
11	9	Leytonstone High Road	7	5	79	8	2	84	6	3	89	6	16	104	0	0	0
12	10	Wanstead Park	4	9	74	4	7	81	3	7	85	13	13	104	0	0	0
13	11	Woodgrange Park	1	12	63	1	8	74	5	14	76	5	19	90	0	0	0
14	12	Barking	0	63	0	0	74	0	0	76	0		90	0	0	0	0

Figure 3.1: A snapshot OnTrain File.

This particular file has records of all trains originating from Gospel Oak to Barking. Here, only five times (trips) have been shown, whereas the actual file has records for 67 trips. To clarify, J

(Joining) means number of passengers joining the train; A (Alighting) means number of passengers leaving the train; and OTD means number of passengers on departure. The reason behind the zeros for the last trip is because the train has been cancelled. Also, cell L14 is empty due to a scanning error of the actual file by the OCR (Optical Character Recognition).

By having this set of three types of information about each trip, the reader is able to identify the station that has the highest/lowest demand. On the other hand, manually checking 67 trips per file is not an easy task.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1														
2		Head Code	Origin Time	Origin	Destination	Platform	Cars	Arr Time	Dep Time	Joining	Alighting	OTA	Late	CAPE
3		2T01	0510	Chingford	Liverpool Street		4	0517	0517	14	0	42		
4		2T03	0525	Chingford	Liverpool Street		4	0532	0532	15	0	38		
5		2T05	0540	Chingford	Liverpool Street		8	0547	0547	25	1	44		
6		2T07	0555	Chingford	Liverpool Street		4	0602	0602	23	0	82		
7		2T09	0610	Chingford	Liverpool Street		8	0617	0617	24	0	114		
8		2T11	0625	Chingford	Liverpool Street		8	0632	0633	49	1	160		
9		2T13	0642	Chingford	Liverpool Street		8	0649	0650	58	2	233		
10		2T15	0657	Chingford	Liverpool Street		8	0704	0705	80	5	285		
11		2T17	0712	Chingford	Liverpool Street		4	0719	0720	72	4	326		
12		2T19	0727	Chingford	Liverpool Street		8	0734	0735	91	2	615		
13		2T21	0742	Chingford	Liverpool Street		8	0749	0750	81	2	665		

Figure 3.2: A snapshot CallingPoint File.

What is shown in the above Figure (3.2) is only 14 lines, while the actual file has 77 lines and in some cases it may have more. To clarify, Head Code refers to the Train ID number for that trip, as the train can have multiple IDs; Origin Time is the time the train left the station it started at, which is the origin station; Cars is the number of coaches making up a train; Arr/Dep Times are the times the train waits at the through station, if it only stops briefly the times will be the same, however at bigger stations the Arrival and Departure would generally be different; Joining means number of passengers joining the train; Alighting means number of passengers leaving the train; and OTA means number of passengers on Arrival. Late is the number of delayed minutes by the train. CAPE refers to whether the train has been cancelled.

The data recoded in this type of survey provides information about the busy times of the day at that station. Thus, that will lead to more cars being added to a train in order to handle the capacity or, in later stages, the expected demand.

	A	B	C	D	E	F	G	H
1	Destination	Head Code	Plat	Dep Time	Cars	Joining	Late	CAPE
2	Heathrow Terminal 4	2T10	11	0512	5	59		
3	Heathrow Terminal 4	2T11	11	0528	5	33		
4	Heathrow Terminal 4	2T14	11	0612	5	334		
5	Reading	2R11	12	0643	3	84		

Figure 3.3: A snapshot of Terminus File

The above Figure has the same terminologies as the one in Figure 3.2. This type of file has two sheets- one for Joining and the other for Alighting. This snapshot is for the Joining sheet. Further to what has been explained about the Terminus scenario in Section 2.2, for the joining sheet, the location of the counting staff is the origin; whereas for the alighting sheet, it is the destination.

The point of the above type of survey is to know the capacity a station can handle per day. According to the client [20], Reading train station has been rebuilt to have more platforms in order to handle the demand.

Having explained the problem, the suggested solution states that instead of solely relying on Microsoft Excel to do the aggregation (manipulation) of data (numbers) to produce the required information, it is thought that a new dedicated visualisation system can do the job automatically and dynamically over the web by producing visual forms so that one graph, for example, can give useful information which can be easily understood. The idea is that the system shall perform a number of jobs simultaneously. For example, it can provide visualisation analysis of every uploaded file, organise relevant projects together, and be a repository for all uploaded records for later analysis.

The visualisation analysis is the big part of the system, and the distinguished reason for building such a system. The system is thought to analyse the data according to some configurations, which will be explained in the next section, handled by the admin. Also, the system is proposed to have an

authentication mechanism (login system) to first prevent any security breach [39, 40] and second to permit access only to those who are allowed by the admin.

Once the system has been built, it is expected that the system will attract benefits to the company, not only in allowing productivity to take place, but also financially [20].

3.2. System Functionality

After a number of meetings with the client, the functionalities required to be implemented in the system have been explained and agreed upon (please see Requirements Catalogue in Appendix C). Besides, the author has suggested some enhancement features to be implemented in the system so that it can attract more benefits to the client, which have also been agreed. In terms of this project, the core features have been fully implemented along with two further enhancements.

3.2.1. Core Features

As stated in Section 1.5, the main deliverable of this project is a web-based visualisation system that satisfies the Requirements Catalogue in line with the minimum requirements. In terms of this project, there are four core requirements, which explain the broad view of the product:

- **Visual Form;**

After an extensive research of the literature [15, 16, 18], a Line Chart has been chosen as the visual form to display the analysed data. To justify this, the chosen visual form is suitable and delivers the necessary information from the data and according to the exact configuration, which will be explained in Sub-section 3.4.3.2. Also, the visual form can be easily understood and articulated.

- **Displaying the Visual Form;**

The way and location of displaying the visual form is important so that the information presented (expressed) in it can be understood correctly, thereby avoiding misinterpretation. In Section 3.4, a clear explanation of the layout of the system along with some snapshots will be given.

- **Friendly and configurable system;**

It has been stated in the literature [4, 24, 25] that in order for the user to benefit from the system's full potential, the system has to be easy to use and friendly. This requires clear and well-presented information in a well-built layout (template) which allows

easy and clear navigation between pages. Besides, the system is built to dynamically accept and configure all three types of files with a well-designed database, which will be explained in Section 3.4.1; therefore, it can handle lots of data with some constraints to avoid clashes and conflicts [41].

- **Evaluation;**

Once the project has been implemented, the evaluation stage takes place. The goal is to know how effective the product is in terms of satisfying the requirements mentioned earlier, as well as how reliable the system is in terms of handling lots of data. In Chapter 5, there will be an explanation of the various ways used in order to better evaluate the system, ranging from code testing to surveying the client.

3.2.2. Enhancement Features

Due to the fact that the author has been interested and willing to accomplish the set targets, a number of enhancement features have been introduced to the client in order to utilise the maximum potential of the system. Due to time constraints, only the first two enhancements, which have been mentioned in Section 1.4, have been implemented.

- **Project Comparison;**

This is a very beneficial aspect of the system. It enables the user not only to analyse the required file, but also to compare between two files (projects). As a result of the comparison, a clear view of the relationship between the two charts can be observed, given that the circumstances of both projects are similar, that is, date, type, station, trip. To illustrate, Season Project comparison is a good example. For example, the user can upload files for Spring 2011 and Spring 2012. Then, he or she chooses one file from each season and compares the analysis results between these two files. One of the benefits of this comparison is to know the difference in capacity for one station between these periods. The term used in the system for this feature is “job”, so a client creates a job and attaches (uploads) many files (mini-projects) relevant to their season.

- **System Management;**

Managing a system is crucial for a variety of reasons, in order keeping the system running in perfect order [42]. A number of basic features have been implemented while building the system; whereas a number have been suggested. It has been suggested that the user should have the ability to archive a project, and after a period

of time, it gets deleted automatically so that there would be a limit so as not overload the database; however, the client did not have a clear view regarding the exact time after which an archived file gets deleted. Thus, this feature has not been implemented and will be considered as a future development. Besides, a mechanism has been implemented that allows the user to update the project's details and to replace its uploaded file. The reason for this feature is to allow the user (admin) to have more control over the system. In addition, a print feature has been implemented so that the user (admin or client) can view the visual form and be able to print it in two formats (pdf or xps), depending on the configuration of the browser.

3.3. Technology

In order for the objectives of this project to be achieved, a number of decisions had to be taken in terms of the programming languages used in building the system.

3.3.1. Programming Language

After reading the literature [43, 44, 45], it has been found that there are three main aspects that influence the choice of the programming language:

- Programmer's skills: it is clear that a programmer will be more productive and innovative if s/he works in a language they know and have mastered, instead of working in a new language [43].
- Platform: the language must be supported by an excellent choice of GUI toolkits. Also, it is far more supportive to allow the platform to rule the programming language, instead of letting the programming language rule the platform [43, 45].
- The language: the programming language has to be easy to learn and understand and must support type checking, turning logic errors into syntax errors [43, 44].

Having stated the above reasons, and due to the client's request, ASP.NET and C#, as its backbone, have been chosen and have been used to build the system.

3.4. Development Sequence

To fully develop the proposed solution, a number of iterations have been undertaken by going through three phases, which are Design, Implementation and Testing, as explained in Section 1.7.1. The aim of these iterations, further to the reasons explained earlier, was to implement a new feature or to enhance previous versions. Figure (3.4) below shows the final version (prototype) of the system that has been achieved. A detailed analysis of the development process will be drawn in the next sub sections. Also, a user guide, given in Appendix F, includes a thorough description of the system supported by screenshots.

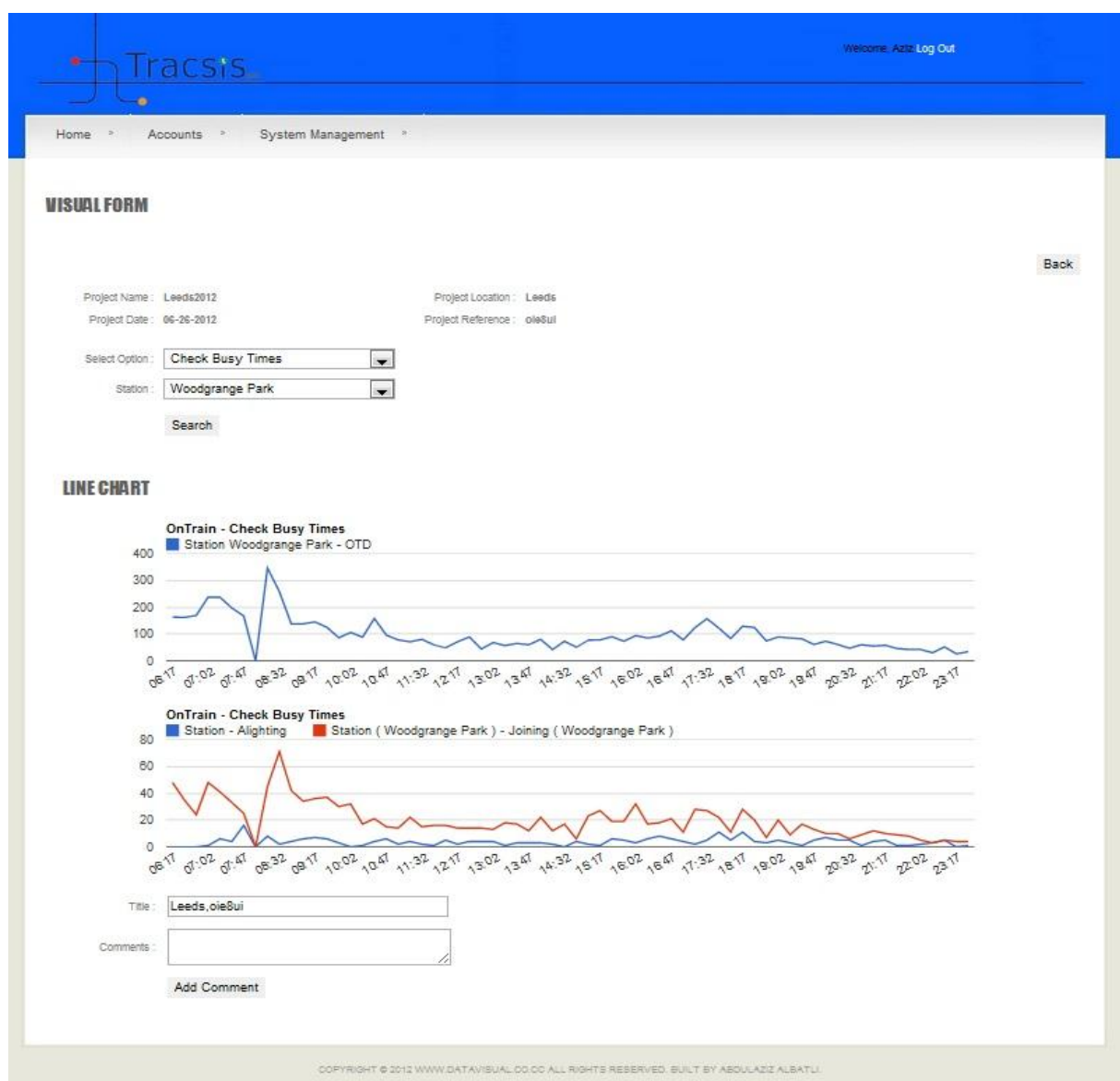


Figure 3.4: A screenshot of the final version of the system

To clearly introduce the processes that have been undertaken, the Figure (3.5) below shows the overall view of the steps that the user will go through (manually or automatically) in order to generate the visual form(s) needed.

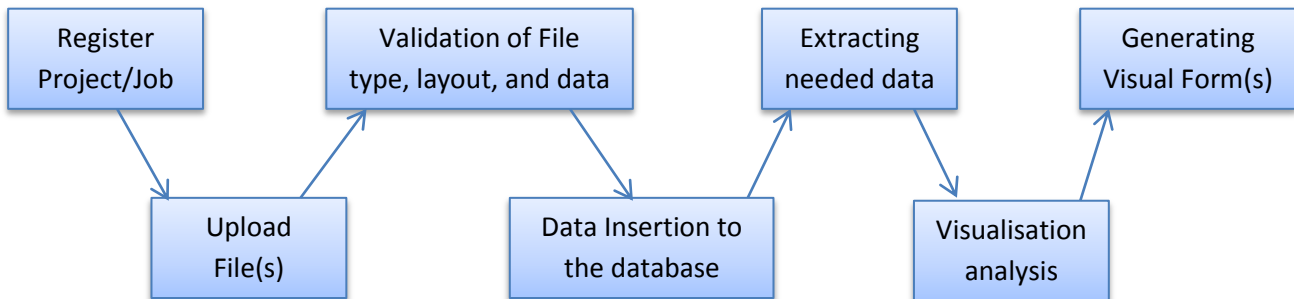


Figure 3.5: Steps to generate visual form(s)

3.4.1. Database and Logic Design

Once the requirements have been understood, the system design in terms of template layout and database design can commence. In order to build the database, a number of system design diagrams have been completed, including Data Flow Diagram, Entity Relationship Diagram and Use Case Diagram. The goal of using these designs is to gain better requirements elicitation [46]. Having said that, the design of the database has been updated a number of times in order to make it fully fit for purpose.

The Figure (3.6) below shows the final database design which is truly the backbone of the system. As can be seen, the database has been configured to accept all three types of files by having four tables, one for each sheet (CallingPoint, OnTrain, TermJoin and TermAlight). To explain, Admin and Client tables are to record their details; Job tables (JobDetails and JobVisualForm) are related to Job details and their generated visual forms. Project tables (Project and VisualForm) are related to project details and their generated visual forms. To clarify, a job consists of a number of mini-projects and is used for project comparison; whereas a project consists of only one file, as explained in Section 3.2.2.

In terms of coding, the Figure (3.7) below shows a screenshot of the code used to insert data into the On Train table using the procedure “insert_OnTrain”, Figure 3.8.

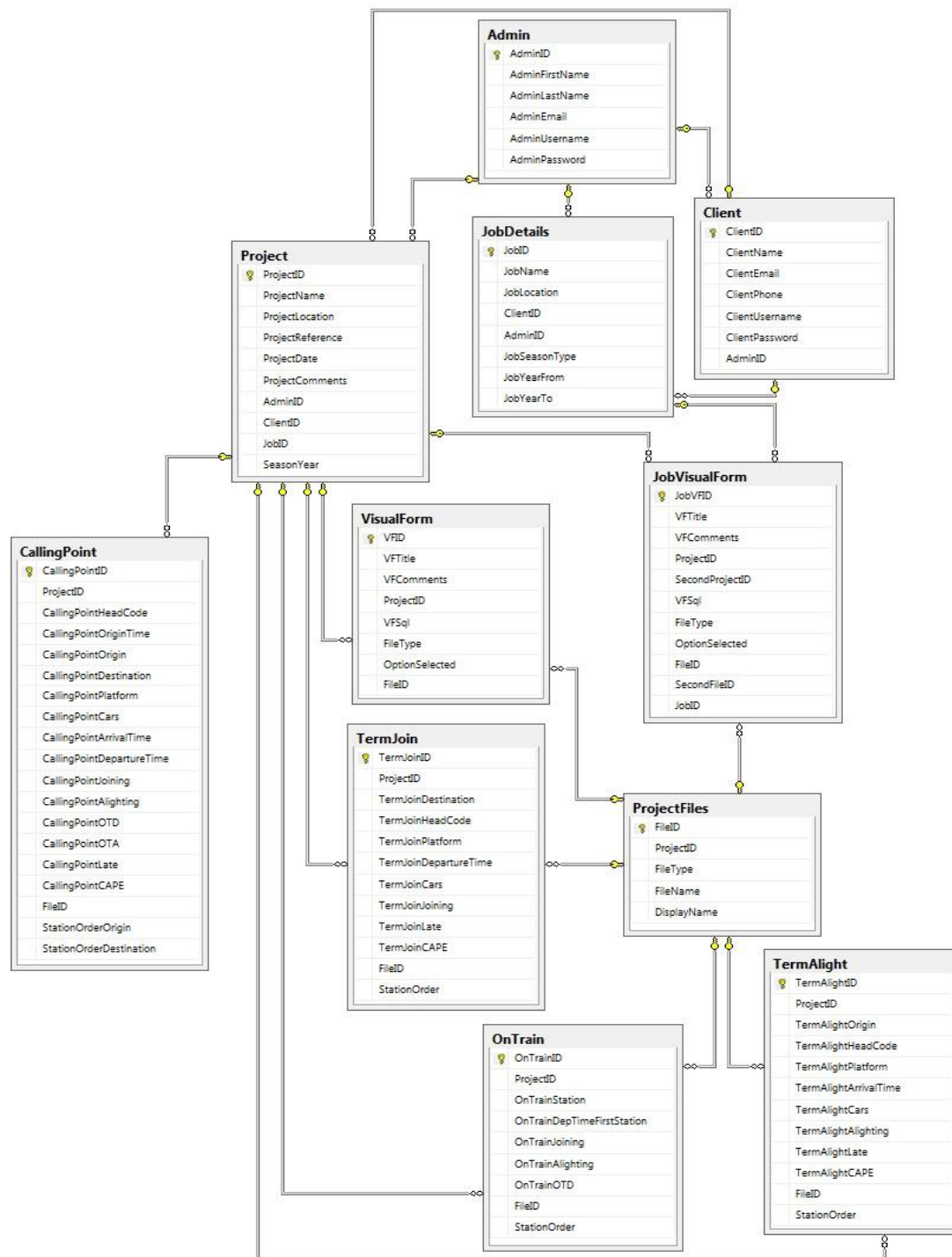


Figure 3.6: The final database design

In terms of the template design, after reviewing a number of templates already built, some sketches have been made taking into consideration the overall requirements and the general audience of the system. According to the client [20], the system has a lot of potential and will be integrated with their system in order to let their clients use it. The main audience of the system are *Tracsis, Plc* and their clients, such as Department of Transport. Thus, it can be seen from the above Figures (1.1 and 3.4) how the design process has evolved.

```

SqlConnection con = new SqlConnection(clsGeneral.GetConnectionString());
SqlCommand cmd;

cmd = new SqlCommand("insert_OnTrain", con);
cmd.CommandType = CommandType.StoredProcedure;
cmd.Parameters.Add("@ProjectID", SqlDbType.Int).Value = ProjectID;
cmd.Parameters.Add("@FileID", SqlDbType.Int).Value = FileID;
cmd.Parameters.Add("@OnTrainDepTimeFirstStation", SqlDbType.Time).Value = OnTrainDepTimeFirstStation;
cmd.Parameters.Add("@OnTrainStation", SqlDbType.VarChar).Value = OnTrainStation.Replace("'", "");
cmd.Parameters.Add("@OnTrainJoining", SqlDbType.Int).Value = OnTrainJoining;
cmd.Parameters.Add("@OnTrainAlighting", SqlDbType.Int).Value = OnTrainAlighting;
cmd.Parameters.Add("@OnTrainOTD", SqlDbType.Int).Value = OnTrainOTD;
con.Open();
cmd.ExecuteNonQuery();
con.Close();

```

Figure 3.7: Insertion Code for On Train Data

```

ALTER PROCEDURE dbo.insert_OnTrain

(
    @ProjectID int,
    @FileID int,
    @OnTrainStation varchar(50),
    @OnTrainDepTimeFirstStation Time(7),
    @OnTrainJoining int,
    @OnTrainAlighting int,
    @OnTrainOTD int)
AS
begin
    Declare @Order int=0;
    Select @Order= IsNull(StationOrder,0) from OnTrain where OnTrainStation=@OnTrainStation and ProjectID=@ProjectID and FileID=@FileID

    if @Order=0
    begin
        Select Top 1 @Order= IsNull(StationOrder,0) from OnTrain where ProjectID=@ProjectID and FileID=@FileID order by StationOrder Desc
        Set @Order =@Order+1
    end

    Insert into OnTrain (ProjectID,FileID,OnTrainStation,StationOrder,OnTrainDepTimeFirstStation,OnTrainJoining,OnTrainAlighting,OnTrainOTD)
    Values(@ProjectID,@FileID,@OnTrainStation,@Order,@OnTrainDepTimeFirstStation,@OnTrainJoining,@OnTrainAlighting,@OnTrainOTD)
end
RETURN

```

Figure 3.8: Insertion Procedure for On Train Data

3.4.2. Admin and Client Areas with Distinguished Roles

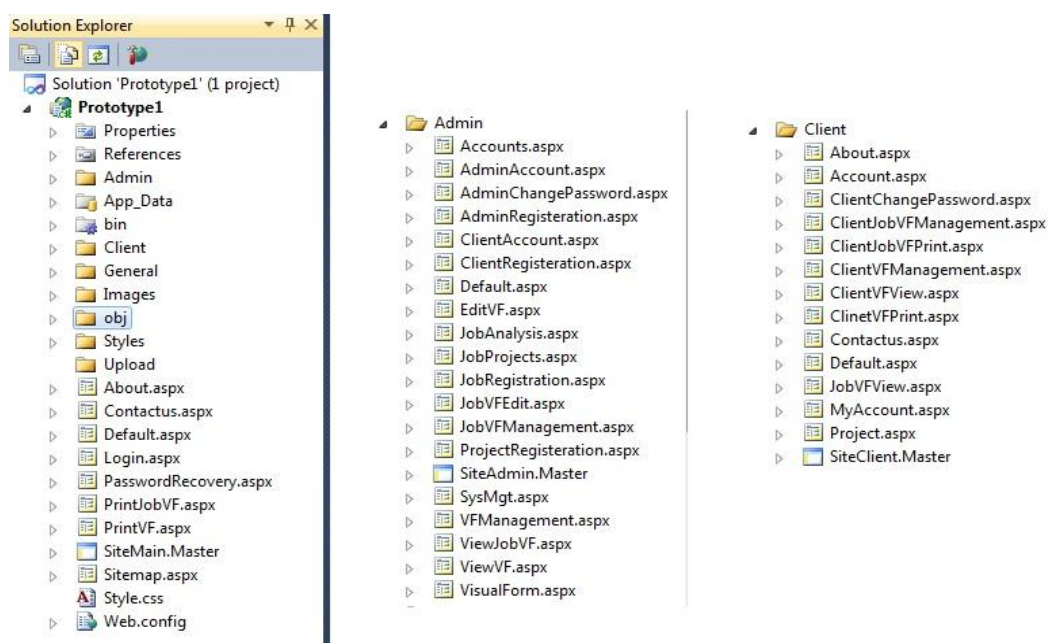


Figure 3.9: System areas

The above Figure (3.9) shows how the system has been divided into three areas, each with its own master page. According to msdc [47] ASP.NET master pages enable a consistent layout for the pages in an application to be developed. Thus, a particular master page outlines the appearance and standard behaviour that are needed for a group of pages in the application. This will allow specific content to be displayed on individual content pages [26, 32]. The benefit is to clearly control the content to be displayed for the Admin, which is different from the content to be displayed for the Client, to guarantee data integrity. In addition, the pages for Admin and Client can only be accessed once he or she goes through a login mechanism. The login system will automatically redirect the user to the relevant master page (area), depending on his or her privilege. Another advantage of having the master pages is that it allows the developer to centralise the common functionality of the linked pages so that updates can be made in just one location [47].

To explain the roles, the Admin basically controls the system. He or she can register/delete users and send them automated emails with their login details. He or she creates projects and jobs and uploads their relevant files. They also control the generation of the visual forms in terms of their configurations. However, the client can mostly view the system. They can see their relevant list of commissioned projects or jobs. They also can view the already generated visual forms. In terms of editing, they can partially update their details but they cannot update their usernames due to security precautions. Also, they can amend the comments related to each visual form.

In terms of coding, the login system manages the redirection of the users (admin and client) to their relevant pages (areas) (see Figure 3.10 below). It first checks the database, using stored procedures, with the username and password entered by the user, and if the validation passes, they will be redirected.

```

if (readeradmin.Read())
{
    Session.Add("adminlogin", "ok");
    Session.Add("AdminID", readeradmin["AdminID"]);

    Response.Redirect("~/Admin/Default.aspx");
    conadmin.Close();
    conclient.Close();
}
else if (readerclient.Read())
{
    Session.Add("clientlogin", "ok");
    Session.Add("ClientID", readerclient["ClientID"]);

    Response.Redirect("~/Client/Default.aspx");
    conadmin.Close();
    conclient.Close();
}

```

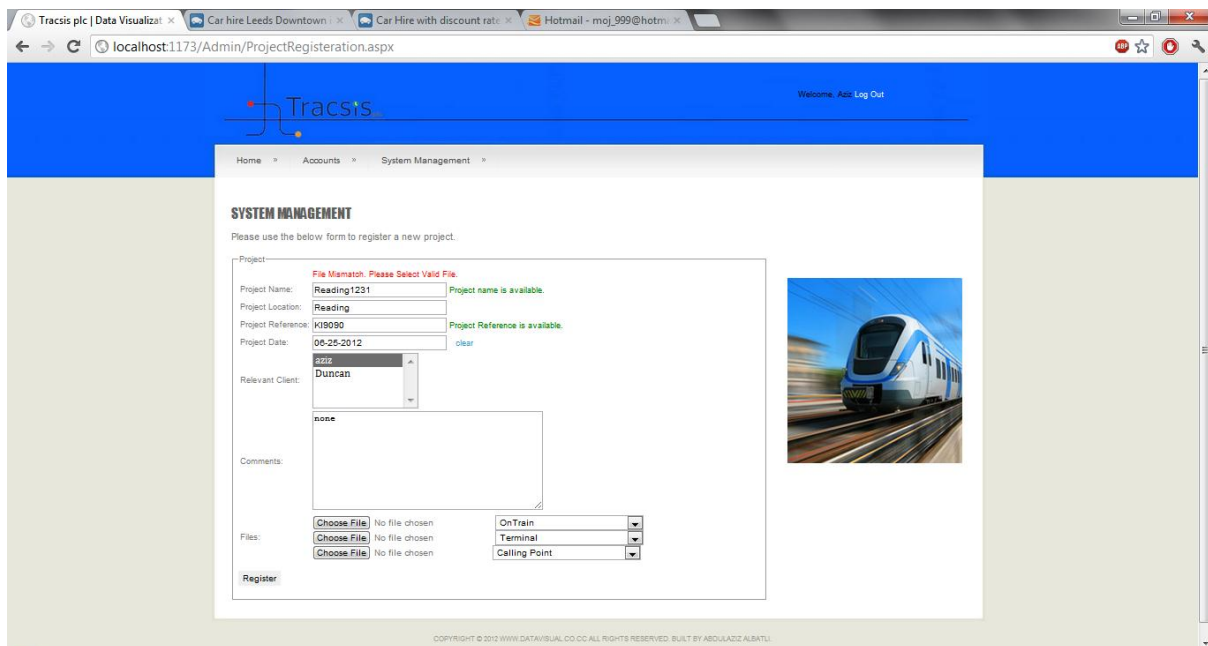
Figure 3.10: Login code

3.4.3. Data Manipulation

As explained earlier in chapter 1 and Section 3.1, the aim of this project is to transform the data from its original format and generate visual forms that clearly express the information. In fact, there are many steps involved before the final chart is created (see Figure 3.5 above).

3.4.3.1. File Validation and Data Insertion

Once the user fills in the project registration form, as seen in Figure 3.11 below, the user has to select the matching file type from the dropdown menu for the file to be uploaded. In case the wrong file type has been chosen, the system will flag a File Mismatch error message. The point of this validation is to ensure data integrity by not storing data in the wrong table, which will most definitely lead to negative consequences. Another validation is the Project Name and Reference Number; while filling in the textbox, the system automatically checks the database and whether the name or number have already been stored or not. Then, depending on the validation, a green or red message is flagged up to notify the user. The advantage of this validation is to prevent duplication, which might not only cause problems in the database, but also confusion for the user.



The screenshot shows a web browser window with the URL `localhost:1173/Admin/ProjectRegistration.aspx`. The page has a blue header with the Tracsis logo and a navigation bar with links to Home, Accounts, and System Management. The main content area is titled "SYSTEM MANAGEMENT" and contains a form for registering a new project. The form includes fields for Project Name (Reading1231), Project Location (Reading), Project Reference (K19090), Project Date (05-25-2012), Relevant Client (Duncan), and a Comments section. There are also three "Choose File" buttons for uploading files, each with a "No file chosen" status. To the right of the form is a dropdown menu for "OnTrain" with options "OnTrain", "Terminal", and "Calling Point". A "Register" button is at the bottom left of the form. A small image of a train is visible on the right side of the form. The footer of the page contains the copyright notice: "COPYRIGHT © 2012 WWW.DATAVISUAL.CO.CC ALL RIGHTS RESERVED. BUILT BY ABOLJAZZ ALBATHI".

Figure 3.11: Project Registration

To clarify, the format for each type of file is fixed and not changeable. So, since the column headings for each file is known, they have been used in the file type validation process as fixed values to be compared against the column headings for each to-be-uploaded file.

Once all the validation processes are passed, data gets stripped from its original format and then inserted in the database. In the next Section, the last three steps stated in Figure 3.5 will be explained.

3.4.3.2. Visualising the Data

This section explains the crucial part of the project, which is visualising the data. Once the user (admin) has created the project, he or she can create the visual forms based on the configuration implemented in the system.

For each file type, there are two analysis options. The reason behind having these options is to allow the admin to have more control over what exactly to generate for their client.

For OnTrain file, the first option is Check Maximum Load per station within a specific period of time, as specified by the user (see Figure 3.12 below). This option enables the user to see in a Line Chart how the line (number of passengers) fluctuates over one day (relevant project date). It can be gleaned from the Chart, not just only the maximum number, but also the minimum.

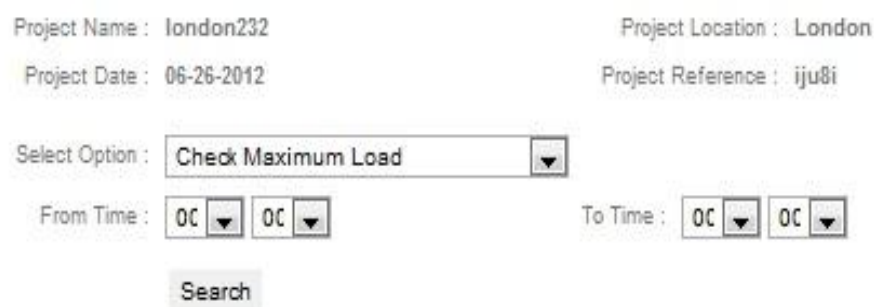


Figure 3.12: OnTrain Option 1 Analysis

The second option is to Check Busy Times per station within one day (see Figure 3.13 below). Because Gospel Oak - Barking train run 67 times per day (see Section 3.1) it would be beneficial to know the busiest time of the day for a station, that is, Crouch Hill. One of the advantages of this result is to increase the number of cars per train to ensure enough seats for all passengers.

Project Name : london232 Project Location : London
 Project Date : 06-26-2012 Project Reference : iju8i

Select Option : Check Busy Times

Station : Please Select

- Please Select
- Barking
- Woodgrange Park
- Wanstead Park
- Leytonstone High Road
- Leyton Midland Road
- Walthamstow Queen's Road
- Blackhorse Road
- South Tottenham
- Harringay Green Lane
- Crouch Hill
- Upper Holloway
- Gospel Oak

© 2012 WWW.DATAVISUAL.CO.UK ALL R

Figure 3.13: OnTrain Option 2 Analysis

For Terminus Files, there are also two options. First, the Peak Loading option, which enables the user to specify the period of time within one day to be analysed along with specifying the station, whether it is an origin or a destination (see Figure 3.14 below). Also, this option provides information about which time is the busiest out of the selected period.

Project Name : london232 Project Location : London
 Project Date : 06-26-2012 Project Reference : iju8i

Select Option : Peak Loading

From Time : 00 00 To Time : 00 00

☒ Origin ☐ Destination Select Station : Please Select

Search

Figure 3.14: Terminus Option 1 Analysis

The second option is Passenger Traffic (see Figure 3.15 below). The output of this analysis is similar to option 1 but without the need to specify the timescale. Instead, it generates a line chart for all trips (times) within one day.

Project Name : london232 Project Location : London
 Project Date : 06-26-2012 Project Reference : iju8i

Select Option : Peak Loading

From Time : 00 00 To Time : 00 00

☒ Origin ☐ Destination Select Station : Please Select

Search

Figure 3.15: Terminus Option 2 Analysis

The last file type, Calling Point, has similar options as well. The first option, Check Busy Times, also enables the user to specify the timescale, see Figure 3.16 below. Also, it enables the origin and destination stations. The advantage of this analysis is to acknowledge the total number of people joining and leaving the specified train (trip). The second option, Check Traffic, is similar to the one related to terminus but without specifying a timescale so that the user can have a broad view of the situation (see Figure 3.17 below).

Figure 3.16: Calling Point Option 1 Analysis

Figure 3.17: Calling Point Option 2 Analysis

3.5. Project Management

Project change is unavoidable in any IT related project. In fact, the most noticeable source of risk in the implementation phase is change to the project. That change often has a direct influence on the time, quality and cost constraints of the software development [48]. In order to manage this project effectively, and to eliminate the chances of project failure, a number of project management documentations have been adopted; please see Appendix E for all relevant documents.

First of all, to manage the implementation phase efficiently, Progress Reports have been produced on a weekly basis in order to keep both the supervisor and the client up to date with the progress. Because WBS has been created in a detailed manner, see Appendix D, it has been really helpful in terms of providing a broad view of the timescale of every task. Besides, to benefit more

from the meeting with the client and the supervisor, and to keep what has been agreed on at every meeting, a Meetings' Minutes document has been produced after every meeting. Lastly, to better manage a project, all possible negative influences on the project have to be considered, such as overestimating the requirements, illness and data loss. Thus, a Risk Management document has been produced, containing suggested actions to be taken to reduce the probability of an accident happening, as well as suggested actions to be taken to reduce the consequences if an accident does occur. For example, three actions have been suggested to reduce probability of data loss incident: first, trying to use efficient memory storage and programs; second, saving the work before shutting down the computer, and third, using multiple backup locations.

All in all, WBS, weekly progress reports, meetings' minutes, and risk management documentations have been key to making the implementation phase successful, as well as the overall project.

4. Results

Having implemented the system, a number of promising results have been achieved. The system is now functioning according to the system requirements (see Appendix C) that have been given by the client, especially the visualisation analysis part of the system. In this chapter, a comparison of the data before and after visualisation analysis will be presented in line with every analysis option of every file type, as mentioned in Section 3.4.3.2 above. Besides, an evaluation of the chosen visual form method (Line Chart) will be given.

4.1. Data Comparison Before and After Analysis

As explained earlier, files that are produced after every survey do not achieve a satisfactory level in terms of explaining the whole picture. In addition, the system can now generate a number of Line Charts per analysis option. For all analysis, the system aggregates the data for the specified station or timescale in terms of J, A, OTA, and/or OTD and plots them on the chart in an already configured manner.

4.1.1. On Train

As it has been explained, the first option, Check Maximum Load, enables the user to determine the station that has the maximum load amongst all stations for the same trip within a specified timescale. The Figure (4.1) below shows the analysis result for the same On Train file mentioned in Section 3.1 and the chosen timescale is 09:00-15:00.

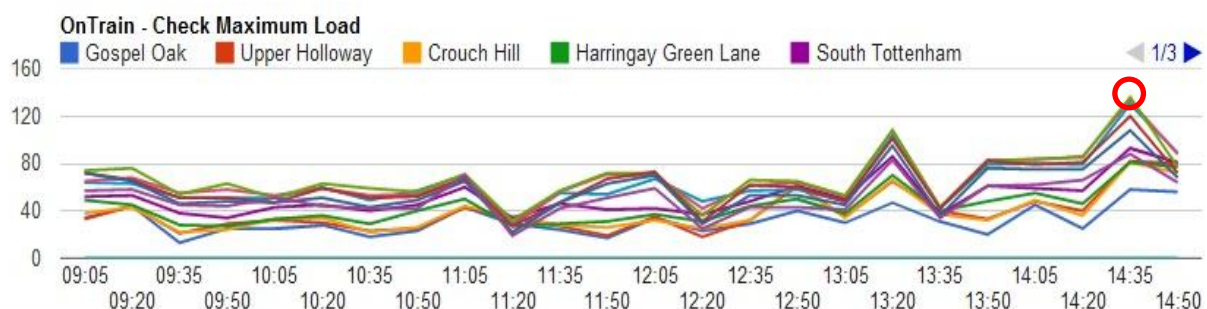


Figure 4.1: On Train Option 1 Result

To clarify, the arrow on the top right corner of the chart is to show the remaining series names. It can be understood from the chart above that Leyton Midland Road station at 14:35 has the maximum load and that Gospel Oak, because it is the origin, has the minimum load.

Moving on, the output of the second analysis option, Check Busy Times, is also in line with the description mentioned earlier; see Figure 4.2 below.

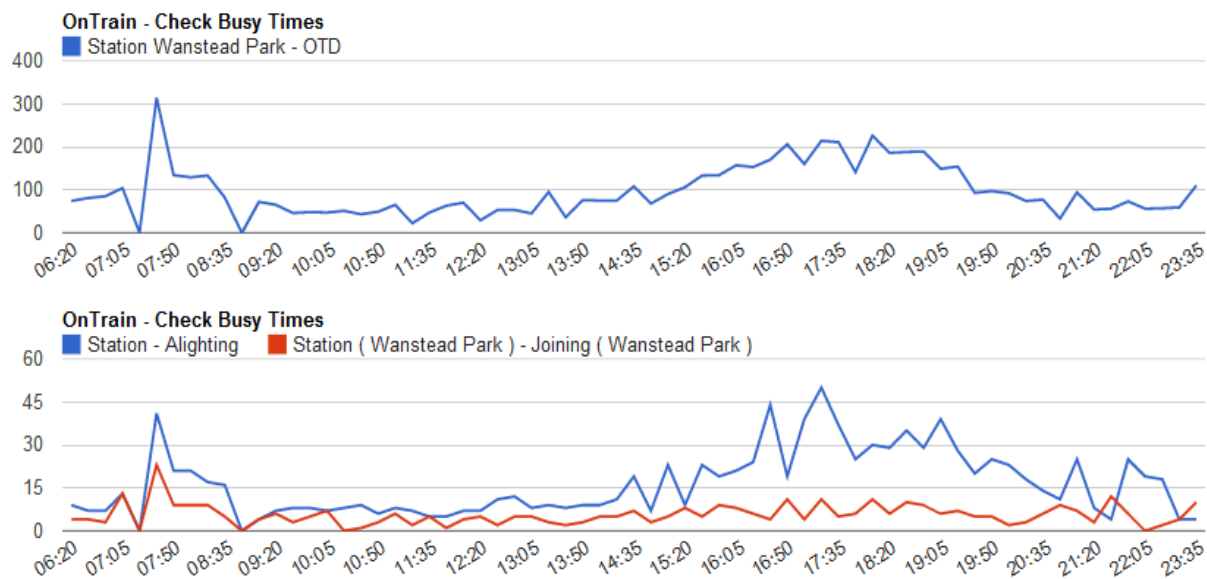


Figure 4.2: On Train Option 2 Result

To explain, the station chosen was Wanstead Park. It can be seen that passenger traffic is high at around early morning and late afternoon.

4.1.2. Terminus

The first analysis option, Peak Loading, has a similar goal, which is to determine the time of the day that is the busiest per a selected station and within a timescale; see Figure 4.3 below.

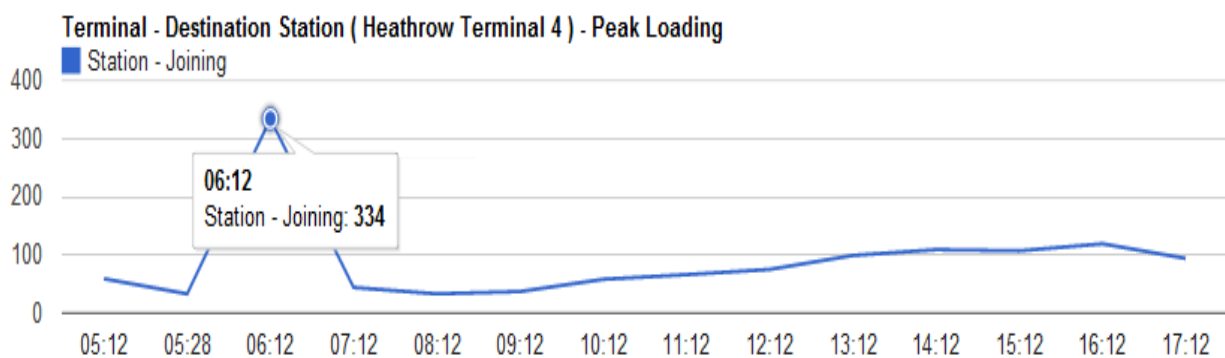


Figure 4.3: Terminus Option 1 Result

The station chosen is Heathrow Terminal 4 as the destination. It can be seen that peak loading is at 06:12.

Furthermore, the second option, Passenger Traffic, gives a broader result, which includes all times of the day with the number of passengers joining or leaving at a station, whether it is an origin or a destination; see Figure 4.4 below.

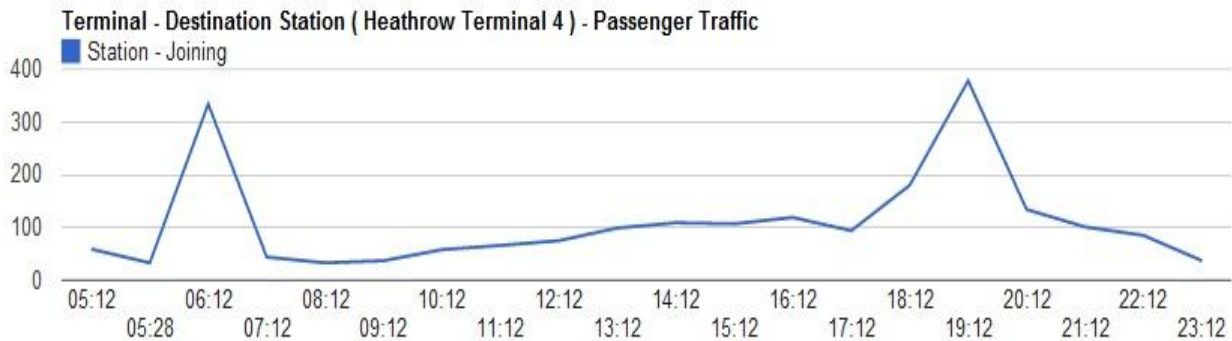


Figure 4.4: Terminus Option 2Result

4.1.3. Calling Point

According to the description mentioned in Section 3.4.3.2, the two options have similar goals for the ones related to the other file types. Check Busy Times allows the user to specify the timescale as well as narrowing down the search by selecting the origin and destination stations. For the Figure (4.5) below, the configuration specified are: Chingford as the origin station and Liverpool Street as the destination station; and the timescale is between 07:00 – 10:00.

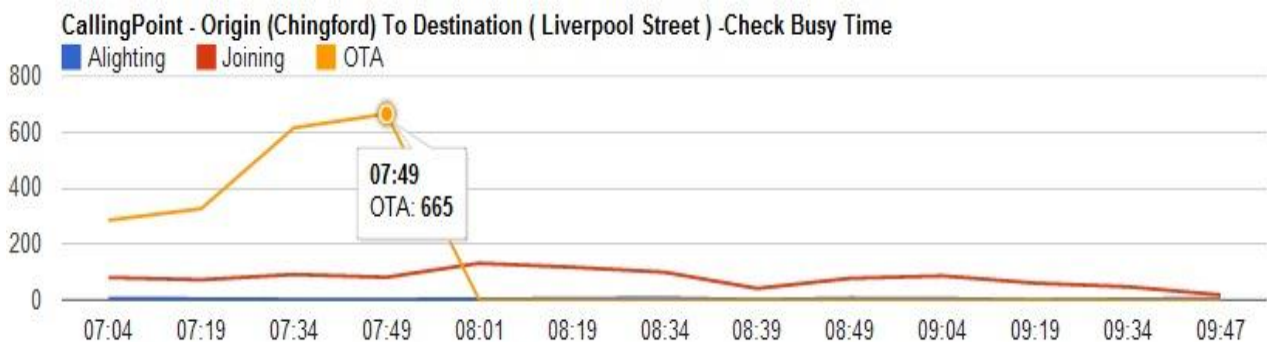


Figure 4.5: Calling Point Option1 Result

The second option, Check Traffic, also provides a wider result by not specifying the timescale. However, it enables the user to have more control by specifying whether the analysis is for *all* trains passing through the current station, the scenario explained in Section 2.2, or for only one *single* train (by choosing the origin and destination). For the Figures (4.6 and 4.7) below, both methods have been illustrated.

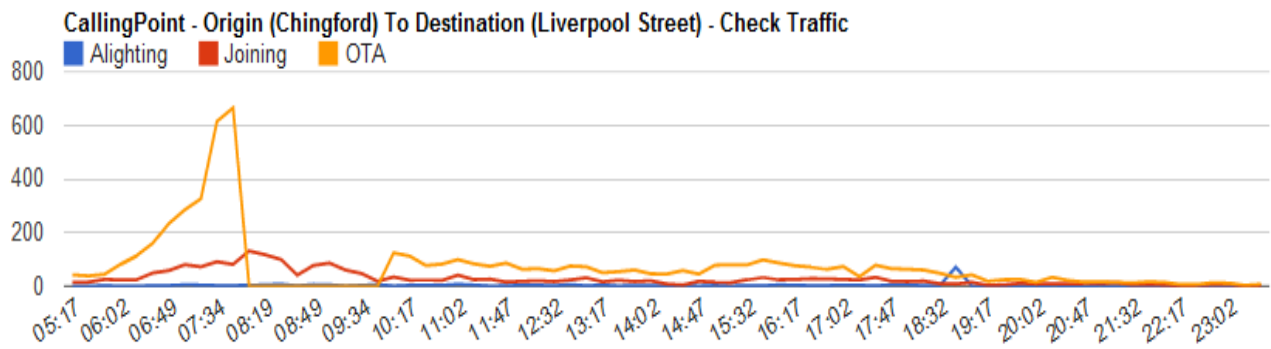


Figure 4.6: Calling Point Option 2 Result, Single

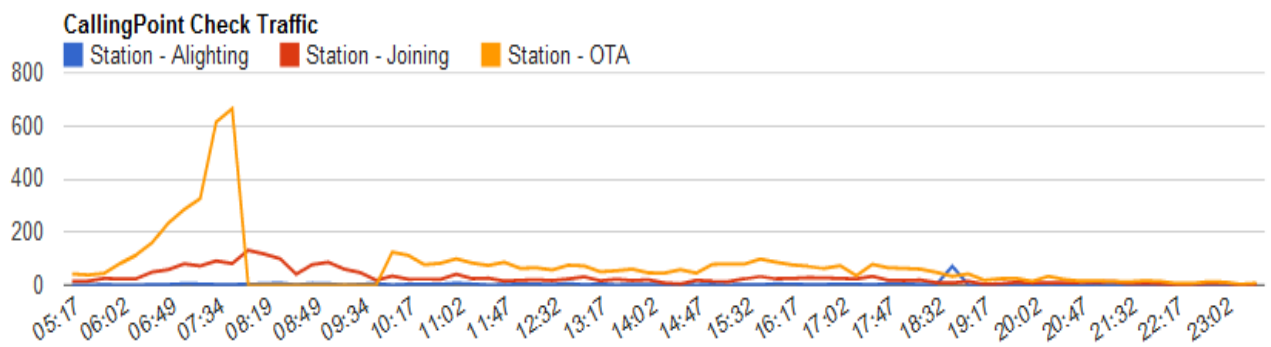


Figure 4.7: Calling Point Option 2 Result, All

In this case, both charts have the same output due to the fact that the file has only one origin station and one destination station. When the ALL option has been selected, for all trains passing through, Joining, Alighting, OTA, and OTD, numbers are added together so that the chart will have three or four lines; each line represents one category.

5. Evaluation

To better evaluate the overall project, the evaluation chapter has been divided into three main sections: Methodology Evaluation, System Evaluation and Results Evaluation. These sections cover the system testing, user acceptance test, and the client's opinion about the system expressed in a survey. The predicted outcomes of this chapter are justifications of how the system has been developed. In addition, in the context of this chapter, it will be explained why some certain choices or decisions have been made so that the level of success can be measured.

5.1. Methodology Evaluation

As mentioned earlier (Section 1.7), the methodology that has been adopted to undertake this project is SDLC (waterfall model), but in an iterative approach. In fact, while adopting this methodology, a number of advantages and limitations have been experienced. As for every other methodology, such as RAD and AGILE, there are advantages and drawbacks which have to be carefully considered by the developer while developing [49]. One of the reasons for adopting a prototyping approach is to handle the disadvantages of the SDLC [50]. The pros and cons that have been faced from the SDLC, combined with the prototyping approach during the implementation of this project, will be explained below.

5.1.1. Pros

Fortunately, the decision of combining SDLC and the prototyping approach has had numerous positive influences on the development of this project. To clarify, in each iterative approach, requirements and different solutions to the issue have been identified and analysed; new methods to resolve the issues have been designed, and as a result a piece of the system was implemented [50, 51]. This helped to eliminate missing functionalities, errors, and omissions in earlier stages of the development cycle, as well as enabling flexible responses to changes. Also, the prototyping approach enforced constant communications and meetings (see Appendix E for documentations of logged meeting minutes with the client). Those meetings have helped in terms of updating and validating the requirements and foreseeing how the next stage looks and functionalities. Besides, the developer has been encouraged during many instances in the development cycle to be innovative, which led to the implementation of a number of further enhancements (see Section 3.2.2). Furthermore, the SDLC has its own advantages [51] in terms of enforcing clear project objectives as well as stable project requirements to be set. That has enabled the progress of the

system to be effectively measured and clear sign-off requirements [52]. Finally, SDLC allows for a full review to be undertaken at the end of every stage to ensure maximum management control.

5.1.2. Cons

Moving on, while there are well suited advantages, there are still some drawbacks that have to be taken into consideration. It has to be mentioned that combining both the SDLC model and the prototyping approach offsets a number of disadvantages if they are implemented separately [49, 52]. In terms of an iterative approach, each iteration phase relies and builds upon the previous one, which eliminates the drawback of starting a whole stage all over again if something goes wrong. Also, sometimes the prototyping approach leads to an incomplete application or system which may not be used as it was initially intended. Also, the iteration approach allows for frequent changes in the requirements, which may force the development of the project to go over budget in terms of time constraints [52]. Besides, system documentation is usually incomplete while the prototyping approach is in place, due to the fact that the focus is on the development of the prototype [51]. The final drawback of the SDLC in the context of this project is that the client can only judge the quality once the development phase is finished. However, that has also been eliminated by the adoption of the iterative approach.

5.2. System Evaluation

In order to evaluate the system properly and thoroughly, it has been suggested by the author that the system evaluation criteria should be divided into three parts, which are Code Testing, User Acceptance Testing, and a Client Survey. This is to ensure the system is running properly with no errors and bugs, whether the requirements specified by the client have been met or not; and to what extent the client believes that the system is reliable as a first prototype.

5.2.1. Code Testing

In fact, code testing has been taking place frequently while developing the system. Actually, the environment used, Microsoft Visual Studio 2010, debugs the system every time before running it [53]. This helps in checking the overall errors and warnings in the system (see Figure 5.1 below). In the context of this project, for every new method written, debugging the system has been run at least once. That has ensured easy correction and identification of errors and bugs, as opposed to debugging the system after a large piece of code is written.

Error List				
0 Errors		0 Warnings		0 Messages
Description	File	Line	Column	Project

Figure 5.1: Debugging Report

5.2.1.1. Manual Testing

Further, to increase the assurance of the dependability of the system, manual testing has been implemented. This is a basic but nonetheless crucial test as it is about checking the status of every button and link of the system. Thus, a manual testing report has been produced which has been included in Appendix G. Also, the Table (5.1) below is a snapshot of a part of the report.

Object of Test	Expected Action	Pass or Fail	Comment
General User			
Home Button	It should redirect the user to the home page.	Pass	
Contact Us Button	It should redirect the user to the contact page.	Pass	
About Us Button	It should redirect the user to the about us page.	Pass	
Site Map Button	It should redirect the user to the site map page.	Pass	

Table 5.1: Manual Testing

After completing the manual testing, it has been confirmed that all physical functionalities (buttons and links) are working as expected.

5.2.2. User Acceptance Test

The user acceptance test is a specific test for ensuring that the requirements specified by the client, see Appendix B, have been correctly implemented and met [54]. It is, in fact, a simple test but its importance relies on the confirmation it delivers, and it is often one of the final stages of developing the system [54, 55]. Also, it is widely used in most engineering projects. The Table (5.2) below explains what the exact requirements that have been met are, from those that have not been.

Requirement	Implemented
Identify and implement a method for visualising the data from files fed into the system, to make meaningful information.	Yes
Identify and implement a suitable way of displaying the graphical forms (i.e. Graphs, Charts, and/or Tables) on the proposed system.	Yes
Implement the system to be friendly (ease of use) and adaptable/configurable to different styles of files (three types have been discussed).	Yes
The system should be web-based.	Yes
The system should be able to take many files and store them in the Database and perform analysis on them.	Yes
Project details include: Name, Location, Reference, Relevant Client, Files and Comment.	Yes
The admin should be the one who creates the visual forms; Client can only view and amend the comments.	Yes
The admin should create the client usernames and send them their login credentials. The clients can update their details except name and username.	Yes
The system should have an authentication process for security (login procedure).	Yes
Identify and implement an appropriate evaluation method for the system.	Yes
There should be an external server for hosting the system.	Yes
Full Testing and Evaluation of the system should be undertaken before handing it to the client.	Yes

Table 5.2: User Acceptance Test

5.2.3. Client Survey

Surveying the client is one of the best ways to know for sure and determine whether the system is dependable and reliable. Thus, an evaluation survey form has been produced, which is included in Appendix H. After deploying the system online (www.tracsisvisual.co.cc), the client has been asked to use the system and evaluate it. Out of the three surveys returned, it can be determined that the client is happy with the system. For example, they have liked the presentation of the data the ability to hover over the graphs and see the breakdown. Also, the options (analysis configurations) that the admin can control while generating a graph have attracted their interest as stated in the surveys. It has been suggested that the system should have the ability to handle more than the three files types. However, since this is a prototype of the system, new feature can be implemented to the system in the future in order to comply with changing requirements.

5.3. Results Evaluation

As has been explained earlier, the visual form that has been chosen for fulfilling the requirements of this project was Line Graphs. Also, it can be seen in sections 3.1 and 4.1 how the charts have provided excellent meaningful visualisation results compared to the data in its original format. In fact, an intensive search of the literature has been carried out, in terms of visual forms, which has led to such satisfactory outcomes. However, the system still has great potential and a number of future development ideas have been suggested which will be explained in the next chapter.

5.3.1. Chosen visual Form

After a number of meetings with the client (see Appendix E) it has been decided that the best visual form that will present the necessary information in a meaningful way is the Line Chart. The reasons behind this decision have been explained in previous chapters. After implementing the charts, it has been noted that for every designated file, the chart showed clear observations of how data changes over time and what the possible pattern is; see Figure 4.1 above. Also, the Line Chart has really proven that it can be easily read and understood [15, 16]. In addition, one of the advantages of the Line Chart is that it clearly shows error value in the data [17].

6. Conclusion and Future Work

To conclude, this dissertation puts forward the process of developing the proposed system which provides a visualisation analysis of raw data, gathered by Tracsis Plc, and fed into it. In order to achieve the aim of this project a number of objectives have been outlined:

- Understand and discuss where, when, how and why data is collected more thoroughly in order to have a better insight when analysing the requirements.
- Understand and discuss web-based systems, including how they are built.
- Understand how to implement the visualisation of data in different ways that are relevant to the type of data being analysed.

In addition, a proposed methodology, the waterfall model, has been undertaken combined with a prototyping approach (iterations). In fact, a number of benefits have been gained by this combination which have been explained and evaluated in chapters 1 and 5, respectively. As explained earlier, the stages that have been included in the iterative approach are: Design Stage, Implementation Stage and Testing Stage.

This chapter has been divided into three main sections in order to reach a comprehensive conclusion. The sections covered are: an outline of the final product, limitations of the product, and future recommendations and developments.

6.1. Outline of the Final Product

Outlining the final product can be best described in two sections: minimum requirements and further enhancements.

6.1.1. Minimum Requirements

This project can be considered to have achieved the agreed minimum requirements, which are:

- Identify and implement a method for visualising the data from files fed into the system, to create meaningful information.
 - Identify the data that is required out of other non-useful data on the files.
 - Convert the data into a database-format and store it onto the database.
 - Identify what is needed to be understood from the visualised data.
- Identify and implement a suitable way of displaying the graphical forms (Graphs, Charts, or Tables) on the web application.

- Implementing the system to be friendly (ease of use) and adaptable/configurable to the different three styles of files in terms of their layout.
- Identify and implement an appropriate evaluation method for the system.

The system meets the above requirements as follows:

- The system can dynamically extract the exact needed data from the uploaded files and store them in a configured manner in the database. Then, the stored data goes through the analysis phase.
- The system then uses the configuration specified by the user to display the appropriate data in the Line Chart.
- The system has been built to be friendly and easy to use. Also, a user guide has been supplied to the client. Besides the system has been configured to accept all three types of files.
- A number of evaluation methods have been undertaken, including Methodology Evaluation, System Evaluation in terms of Code Testing, User Acceptance Testing and Client Survey, and finally, Results Evaluation.
- The system has been deployed online and a dedicated domain name has been set, which is: www.tracsisvisual.co.cc.

6.1.2. Further Enhancements

In addition to meeting the above requirements, other functions were developed and implemented in order to enable maximum utilisation of the system. The enhancement features include the following:

- **Project Comparison;**

This feature enables the user not only to analyse the required file, but also to compare between two files (projects). As a result of the comparison, a clear view of the relationship between the two charts can be observed, given that the circumstances of both projects are similar, that is, date, type, station, trip.

- **System Management;**

A mechanism has been implemented which allows the user to update the project's details and to replace its uploaded file(s). The reason for this feature is to allow the user (admin) to have more control over the system. In addition, a print feature has been implemented so that the user (admin or client) can view the visual form and be able to

print it in two formats (pdf or xps), depending on the configuration of the browser. Also, the admin can send already configured emails to other system users which have their login details.

6.2. Limitations

Even though the system has been implemented to the level that was expected by the client, a number of drawbacks and limitations have been noted. First, the system design (webpage template) does not meet the level of professionalism as expected (see Figure 3.11 above). It was deliberated and decided to use the given timescale to focus on the implementation side of the system in order to meet the expectations in terms of functionality. Second, even though the system can accept different types of files and performs numerous analyses on them, it seems that it only does one job (visualisation). It would be more attractive if it was part of bigger software which can then be sold, on a membership basis, to clients. Thirdly, although files are small in size (~45 KB), there has to be a maximum limiting mechanism in terms of storing data onto the database in order to maintain system performance. This feature has not been implemented and has been justified in Section 3.2. Lastly, a bug has been discovered during the client evaluation stage. Its presence is due to some empty values in the actual excel files. To justify, the system has been configured in a way that it expects that all values (cells) not to be empty as per the client's requirements in the initial stages. To overcome this issue, the system could easily be configured to add a default value to any cell that is empty.

6.3. Future Work and Recommendations

During the development phase, and as a result of the feedback gained from every meeting with the client, the basis for future work has been set in order to improve and enhance the system.

First of all, the database of the system is its backbone and if it is not maintained properly, the system may lose its creditability. One of the important features, which has been proposed earlier, is to have a time limit, after which all data in the database is cleaned (archived). Since every file is crucial for every commissioned project, it is suggested that no file is deleted but is archived instead. This would enable the company to have an archive of past electronic, cleansed, and ready-for-analysis data for later analysis. Besides, by doing that, it would ensure maximum system-performance is maintained. Another crucial enhancement for the system is to have forward-thinking clickable-charts in place. It is noted that the line chart may trigger more questions that need to be

answered. Thus, once the chart has been generated, it can be implemented in a format that is clickable, so it can generate another chart once clicked. For example, Figure 4.1 above shows the station that has the maximum load amongst all stations for the same trip within a specified timescale. Therefore, the user may click on the red circle to get another or multiple visual form(s) generated to show data for all trains that passed through that specific station at that time on that day.

7. Bibliography

- [1] Tracsis Plc (n.d.) **About Us**. [Internet]. Available at: < <http://www.tracsis.com/about-tracsis.htm> > [Accessed 18th July 2012].
- [2] Tracsis Plc (n.d.) **Passenger Counts**. [Internet]. Available at: < <http://www.tracsis.com/passenger-counting.htm> > [Accessed 18th July 2012].
- [3] Bocij, P. et al (2008) **Business Information Systems: Technology, Development and Management for the E-Business**. 4th ed. England, Pearson Education Limited.
- [4] Casteleyn, S (2009) **Engineering Web Applications**. Berlin, Springer-Verlag.
- [5] Rosenblatt, H. and Shelly, G., (2010). **Systems Analysis and Design**. 8th ed. USA, Course Technology.
- [6] Leffingwell, D. and Widrig, D. (2003) **Managing Software Requirements: a Use Case Approach**. 2nd ed. USA, Pearson Education, Inc.
- [7] Rad, F. & Cioffi, F. (2004) Work and Resource Breakdown Structures for Formalized Bottom-Up Estimating. **Cost Engineering** [Internet], 46(2), pp. 31-37. Available from: < [web.ebscohost.com.wam.leeds.ac.uk](http://web.ebscohost.com/wam.leeds.ac.uk) > [Accessed 30th March 2012].
- [8] Doest, L. et al (2006) Personal Goal Facilitation through Work: Implications for Employee Satisfaction and Well-Being. **Applied Psychology: An International Review** [Internet], 55(2), pp. 192-219. Available from: < [web.ebscohost.com.wam.leeds.ac.uk](http://web.ebscohost.com/wam.leeds.ac.uk) > [Accessed 30th May 2012].
- [9] Telea, A. (2008) **Data Visualization: Principles and Practice**. India, A K Peters, Ltd.
- [10] Bulpitt, A. (2012) **Envisioning Information**. Lecture notes for (COMP5390M) Techniques for Knowledge Management. University of Leeds.
- [11] Williams, J. et al. (1995) Visualization. **Annual Review of Information Science and Technology (ARIST)** 30(1), pp. 161-217.
- [12] Spence, R. (2007) **Information Visualization**. 2nd ed. UK, Pearson Education Ltd.
- [13] Fry, B. (2007) **Visualizing Data**. USA, O'Reilly Media, Inc.
- [14] Ken Brodlie, et al. (2004) Distributed and Collaborative Visualization. **Eurographics** [Internet], volume 0 (1981), pp.1-29. Available from: <http://www.comp.leeds.ac.uk/vvr/gViz/publications/CGF_dcvstar.pdf > [Accessed 25th May 2012].

- [15] Ministry of Education, New Zealand (2011) **Tables and Graphs** [Internet]. Available from: <<http://arb.nzcer.org.nz/supportmaterials/tables.php#1>> [Accessed 19th July 2012].
- [16] Jelen, B. (2011) **Charts and Graphs: Microsoft Excel 2010**. USA, Que Publishing.
- [17] Gott, R., Duggan, S. (2003) **Understanding and Using Scientific Evidence: How to Critically Evaluate Data**. London, Sage Publications.
- [18] Sild, M. (2007) **Describing Graphs** [Internet]. Available from: <<http://www.slideshare.net/meerisild/describing-graphs>> [Accessed 19th July 2012].
- [19] Bengler, W. et al. (2009) Beyond the Visualization Pipeline: The Visualization Cascade. **5th High-End Visualization Workshop** [Internet]. Available from: <<http://sciviz.cct.lsu.edu/papers/2009/VizCascade.pdf>> [Accessed 1st June 2012].
- [20] Moss, D. (2012) **Representative of Tracsis, Plc. as a Client**.
- [21] Han, J. et al. (2012) **Data Mining: Concepts and Techniques**. 3rd ed. USA, Elsevier Inc.
- [22] Bramer, M. (2007) **Principles of Data Mining**. UK, Springer-Verlag Ltd.
- [23] Bulpitt, A. (2012) **Knowledge Discovery/Data Mining**. Lecture notes for (COMP5390M) Techniques for Knowledge Management. University of Leeds.
- [24] Chong, S and others (2009) Building Secure Web Applications with Automatic Partitioning. **Communications of the ACM** [Internet], 52(2) February, pp. 79-87. Available from: <<http://web.ebscohost.com>> [Accessed 5th June 2012].
- [25] Eernisse, M., (2006) **Build your Own Ajax Web Applications**. Victoria, SitePoint Pty, Ltd.
- [26] Ruvalcaba, Z. (2004) **Build your own ASP.NET Website Using C# & VB.NET**. Australia, SitePoint Pty. Ltd.
- [27] Albatli, A. and Alzamil, I. (2011) A Prototype of An Automated Live Weather Interpolation System Using A Web Application. **7th International Conference on Next Generation Web Services Practices (NWeSP), IEEE**. [Internet] Available from: <<http://ieeexplore.ieee.org>> [Accessed 10th June 2012].
- [28] Nations, D., (2010) **Web Applications**. [Internet] Available from: <http://webtrends.about.com/od/webapplications/a/web_application.htm> [Accessed 5th June 2012].
- [29] Blodget, H. (2011) Amazon's Cloud Crash Destroyed Many Customers' Data. **MSNBCB Technolog** [Internet] Available from: <<http://www.technolog.msnbc.msn.com>> [Accessed 5th June 2012].

- [30] Aljawarneh, S. and Alkhateeb, F., (2009) Design and Implementation of New Data Validation Service (NDVS) Using Semantic Web Technologies in Web Applications. **World Congress on Engineering 2009** [Internet], (Volume 1) pp. 179-184. Available from: <<http://web.ebscohost.com>> [Accessed 6th June 2012].
- [31] Cherkasova, L., et al (2010) DP+IP = Design of Efficient Backup Scheduling. **International Conference on Network and Service Management (CNSM)** [Internet]. Available from: <<http://ieeexplore.ieee.org>> [Accessed 6th June 2012].
- [32] Chris Hart and others (2006) **Beginning ASP.NET 2.0**. Indiana, Wiley Publishing, Inc.
- [33] Holzner, S. (2008) **PHP: The Complete Reference**. USA, McGraw Hill Co.
- [34] Gilmore, W. (2011) **Beginning PHP and MySQL: From Novice to Professional**. 4th ed. USA, Apress.
- [35] Grannell, G., (2007) **The Essential Guide to CSS and HTML Web Design**. USA, Apress.
- [36] Xie, D., and Liberty, J. (2008) **Programming C# 3.0**. 5th ed. USA, O'Reilly Media, Inc.
- [37] Albahari, B., and Albahari J., (2007) **C# 3.0 in a Nutshell**. USA, O'Reilly Media, Inc.
- [38] Duckett, J. (2004) **Beginning Web Programming with HTML, XHTML, and CSS**. Canada, Wiley Publishing, Inc.
- [39] Teodoro, N.; Serrao, C., (2011) Web Application Security: Improving Critical Web-Based Applications Quality Through In-Depth Security Analysis. **2011 International Conference on Information Society (i-Society)** [Internet], pp.457-462, 27-29 June. Available from: <ieeexplore.ieee.org> [Accessed 21st July 2012].
- [40] Madan, S., (2010) Security Standards Perspective to Fortify Web Database Applications from Code Injection Attacks. **International Conference on Intelligent Systems, Modelling and Simulation (ISMS)** [Internet], pp.226-230, 27-29 Jan. Available from: <ieeexplore.ieee.org> [Accessed 21st July 2012].
- [41] Camolesi, L. (2004) Survivability and applicability in database constraints: temporal boundary to data integrity scenarios. **Proceedings. ITCC 2004. International Conference on Information Technology: Coding and Computing** [Internet], vol.1. pp. 518- 522, 5-7 April. Available from: <ieeexplore.ieee.org> [Accessed 22nd July 2012].
- [42] Gal, A. and Mylopoulos, J., (2001) Toward Web-based application management systems. **IEEE Transactions on Knowledge and Data Engineering** [Internet], 13 (4), pp.683-702, Jul/Aug. Available from: <ieeexplore.ieee.org> [Accessed 22nd July 2012].
- [43] Britton, C. (2008) Choosing a Programming Language. **MSDN** [Internet]. Available from: <<http://msdn.microsoft.com>> [Accessed 22nd July 2012].

- [44] Spinellis, D. (2006) Choosing a programming language. **Software, IEEE** [Internet], 23 (4), pp.62-63, July-Aug. Available from: <ieeexplore.ieee.org> [Accessed 22nd July 2012].
- [45] Naiditch, D. (1999) Selecting a programming language for your project. **Aerospace and Electronic Systems Magazine, IEEE** [Internet], 14 (9), pp. 11-14. Available from: <ieeexplore.ieee.org> [Accessed 22nd July 2012].
- [46] Jilanic, A. et al (2008) Formal Representations of the Data Flow Diagram: A Survey. **Advanced Software Engineering and Its Applications 2008** [Internet], pp.153-158, 13-15 Dec. Available from: <ieeexplore.ieee.org> [Accessed 23rd July 2012].
- [47] Msdn (2012) **ASP.NET Master Pages** [Internet]. Available from: <<http://msdn.microsoft.com/en-us/library/wtxbf3hh.aspx>> [Accessed 23rd July 2012].
- [48] Hu, E., and Liu, Y. (2008) IT Project Change Management. **International Symposium on Computer Science and Computational Technology'08**. vol.1, pp.417-420, 20-22 Dec. Available from: <ieeexplore.ieee.org> [Accessed 24th July 2012].
- [49] Bocij, P., et al (2008) **Business Information Systems - Technology, Development and Management for the e-Business**. 4th ed. England, Prentice Hall.
- [50] Coopridner, J.G. and Henderson, J.C. (1990) Technology-process fit: perspectives on achieving prototyping effectiveness. **Proceedings of the Twenty-Third Annual Hawaii International Conference on System Sciences** [Internet], vol.3, pp.623-630, 2-5 Jan. Available from: <ieeexplore.ieee.org> [Accessed 29th July 2012].
- [51] Stair, R., et al (2010) **Principles of Information Systems**. 9th ed. USA, Course Technology.
- [52] Chen, A. and Buchs, D. (2006) Generative Business Process Prototyping Framework. **Seventeenth IEEE International Workshop on Rapid System Prototyping**, pp.140-148, 14-16 June. Available from: <ieeexplore.ieee.org> [Accessed 29th July 2012].
- [53] Moore, A., (2010) **Visual Studio 2010 All-in-One For Dummies**. NJ, Wiley Publishing, Inc.
- [54] Grood, D., (2008) **Test Goal: Result-Driven Testing**. Leiden, Collis B.V.
- [55] Torchiano, M., et al (2007) Talking tests: a Preliminary Experimental Study on Fit User Acceptance Tests. **First International Symposium on Empirical Software Engineering and Measurement, 2007**. ESEM, pp.464-466, 20-21 Sept. Available from: <ieeexplore.ieee.org> [Accessed 31st July 2012].
- [56] Chen, M. et al. (2009) Data, Information, and Knowledge in Visualization. **Computer Graphics and Applications, IEEE** [Internet], 29 (1), pp.12-19 Jan. Available from: <ieeexplore.ieee.org> [Accessed 6th June 2012].

- [57] Diehl, S. (2007) **Software Visualization: Visualizing the Structure, Behaviour, and Evolution of Software**. Berlin, Springer-Verlag.
- [58] Rao, C. et al (2005) **Data Mining and Data Visualization**. UK, Elsevier B. V.

8. Appendices

8.1. Appendix A (Reflection)

Honestly, this project has been the most challenging, yet the most exciting, part of my MSc degree. The project has allowed a number of learning opportunities to be had, both in terms of technical and non-technical skills. To develop this project, a number of unpredicted obstacles have been tackled. There are a number of recommendations and suggestions for my fellow MSc students that need to be stated.

Project management is a crucial key for a project to be successful. Formulating a strong and well prepared project plan and trying as much as possible to stick to it will surely increase the level of achievement. The goal of setting out a project plan with clear and definite milestones will help and ensure that every stage of the project development has adequate time. It has to be carefully considered that the given timescale is not long, and without clear preparation, going over it would be unavoidable which might lead to not achieving the expected outcomes. Besides, this project has taught the author to produce a number of management documents, all for the sake of not losing control of the project. Since this project is attached to an external company, it was wise to maintain a log, and minutes of every meeting undertaken so that both the client and developer can keep track of the feedback given and not underestimate them. In addition, a weekly report of the implementation phase has helped in dividing the larger parts into smaller ones to be easily implemented, as well as keeping the supervisor up to date with the progress. One thing that has to be mentioned is that the project plan will most certainly change due to unpredicted circumstances, either through overestimating or underestimating a task, and that has happened for the project plan of this project.

The methodology used for this project was a combination of both the waterfall model and an iterative approach. Thorough research has been undertaken in order for that decision to be made. To reflect, the combination has really helped in foreseeing the whereabouts of risks, both in terms of coding, that is, bugs, and in terms of sticking to the project plan, that is, underestimating a task. It has been helpful to divide the work into small stages that require a separate focus and concentration. Then for each stage, the big tasks have also been developed in stages (iterations). One of the clear benefits was that the client was able to see how the product would look before finishing it, both in terms of its functionalities and design. Basically, the system was being watched as it grew, function by function, which has really pleased the client.

Furthermore, the first step undertaken was to agree on the requirements with the client. This project has provided the chance for the developer to learn one thing that has not been covered in the MSc course, which is requirements analysis. In order for the requirements to be gathered and analysed, a number of interviews had to be carried out, which is another skill that has been gained. Those meetings have enabled the author to better understand what was required in a regular language, not a programming one. Besides, it has been thought that meeting the client on a regular basis would have been good for both keeping him updated and refreshing (changing) the requirements.

All in all, I have really enjoyed the project from the start. Before deciding on doing the project, both the challenge element and the excitement element were attached to it, otherwise the project would have lacked the enjoyment part. A number of learning opportunities have been gained both in terms of new concepts and techniques. Without a doubt, it has been a positive experience which has enabled the utilisation of the knowledge gained during the MSc course.

8.2. Appendix B (Interim Report Feedback)

Supervisor's comments on the Interim Report

This project is interesting with an industrial client. A clear high-level system design showing its components and their relationships would be needed. A good description of the rail passenger counting scenarios would also be important to understand the data to be visualised and the purpose of the surveys being commissioned.

Assessor's comments on the Interim Report

This is a well written report on what looks to be an interesting project.

The Background reading on the methodology is rigorous and well presented.

Although the project may not be in its implementation stage, I would have expected some information on the format of the data, the steps that are needed to process them. I would also have expected some information about how the data is going to be visualised, for example design sketches, etc.

Please allow for more than two weeks of writing. Start earlier.
Good luck.

Marc de Krom

8.3. Appendix C (Requirements Catalogue)

8.3.1. Original

- Identify and implement a method for visualizing the data from files fed into the system, to make meaningful information.
- Identify and implement a suitable way of displaying the graphical forms, i.e. (Graphs, Charts, and/or Tables) on the proposed system.
- Implementing the system to be friendly (ease of use) and adaptable/configurable to different styles of files (three types have been discussed).
- The system should be web-based.
- The system should be able to take many files and store them in the Database and do analysis on them.
 - o For example, analysis could be Average/Total number of passengers on a particular line/particular train/particular season.
- The system should have an authentication process for security (login procedure).
- Identify and implement an appropriate evaluation method for the system.
- Have an external server for hosting the system.
- Full Testing and Evaluation of the system should be undertaken before handing it to the client.

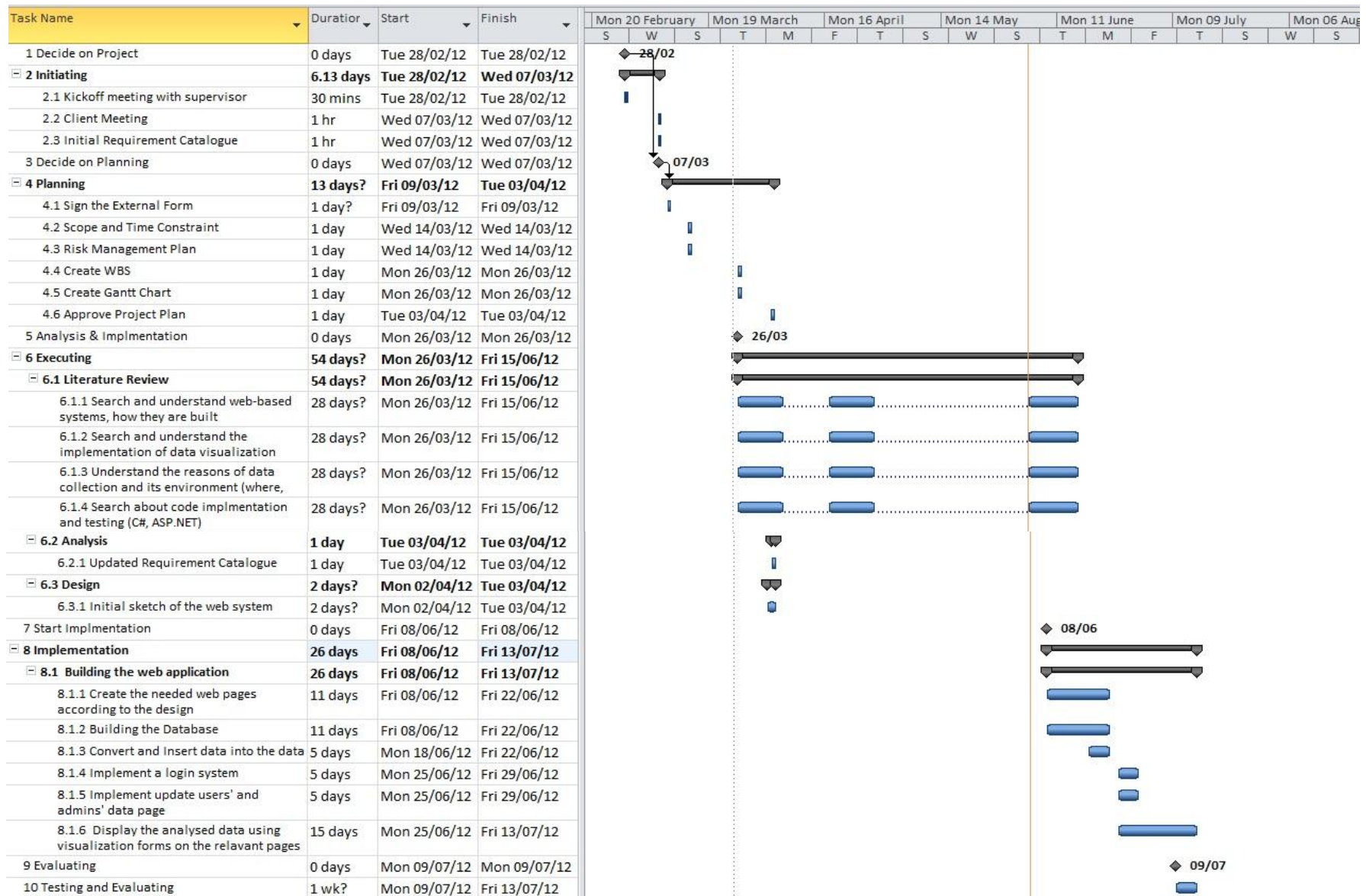
8.3.2. Updated

- Identify and implement a method for visualizing the data from files fed into the system, to make meaningful information.
- Identify and implement a suitable way of displaying the graphical forms, i.e. (Graphs, Charts, and/or Tables) on the proposed system.
- Implementing the system to be friendly (ease of use) and adaptable/configurable to different styles of files (three types have been discussed).
- The system should be web-based.
- The system should be able to take many files and store them in the Database and do analysis on them.
 - o For example, analysis could be Average/Total number of passengers on a particular line/particular train/particular season.
- The details required for a project:
 - o Name
 - o Location
 - o Reference
 - o Relevant Client
 - o File(s) to upload
 - o Comment
- The admin should be the one who creates the visual forms; Client can only view and amend the comments.
- The admin should create the clients usernames and send them their login credentials. The clients can update theirs details except name and username.

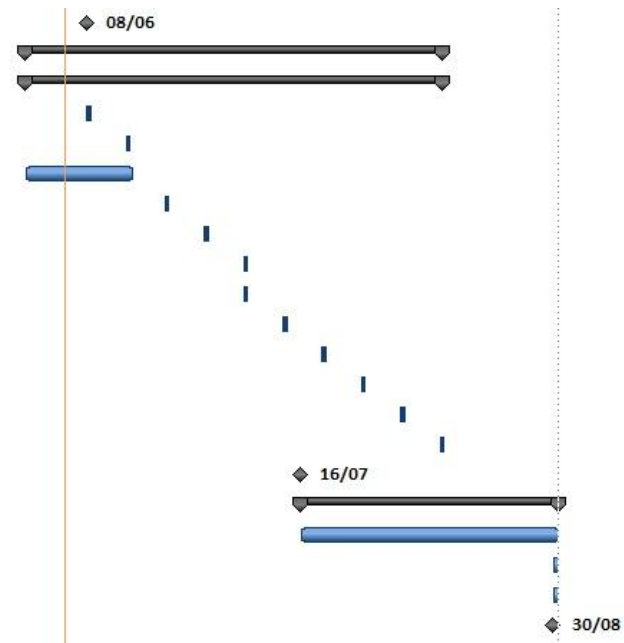
- The system should have an authentication process for security (login procedure).
- Identify and implement an appropriate evaluation method for the system.
- Have an external server for hosting the system.
- Full Testing and Evaluation of the system should be undertaken before handing it to the client.

8.4. Appendix D (WBS)

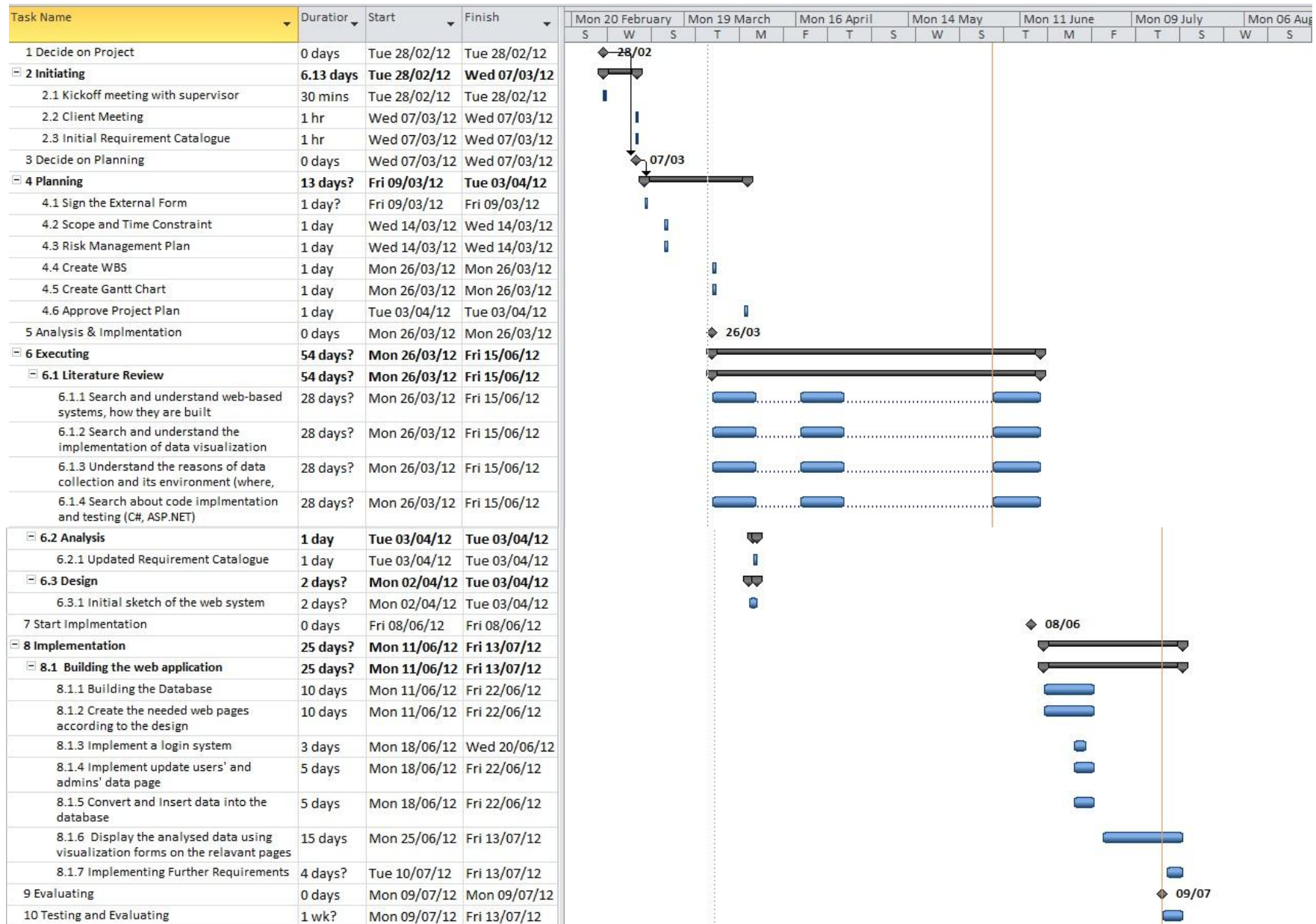
8.4.1. Original



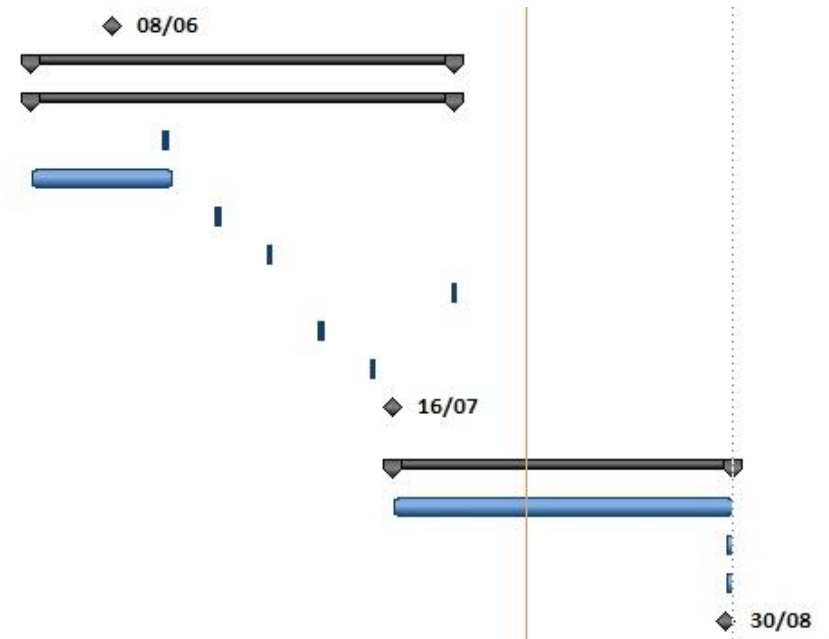
11 Monitoring	0 days	Fri 08/06/12	Fri 08/06/12
▢ 12 Controlling	54.13 days	Mon 28/05/12	Fri 10/08/12
▢ 12.1 Progression Status	54.13 days	Mon 28/05/12	Fri 10/08/12
12.1.1 Progress Report 1	1 hr	Fri 08/06/12	Fri 08/06/12
12.1.2 Progress Report 2	1 hr	Fri 15/06/12	Fri 15/06/12
12.1.3 Writing Interim Report	15 days	Mon 28/05/12	Fri 15/06/12
12.1.4 Progress Report 3	1 hr	Fri 22/06/12	Fri 22/06/12
12.1.5 Progress Report 4	1 hr	Fri 29/06/12	Fri 29/06/12
12.1.6 In Progress Presentation	2 hrs	Fri 06/07/12	Fri 06/07/12
12.1.7 Progress Report 5	1 hr	Fri 06/07/12	Fri 06/07/12
12.1.8 Progress Report 6	1 hr	Fri 13/07/12	Fri 13/07/12
12.1.9 Progress Report 7	1 hr	Fri 20/07/12	Fri 20/07/12
12.1.10 Progress Report 8	1 hr	Fri 27/07/12	Fri 27/07/12
12.1.11 Progress Report 9	1 hr	Fri 03/08/12	Fri 03/08/12
12.1.12 Progress Report 10	1 hr	Fri 10/08/12	Fri 10/08/12
13 Preparing the final project and the Dissertati	0 days	Mon 16/07/12	Mon 16/07/12
▢ 14 Closing	34 days	Mon 16/07/12	Thu 30/08/12
14.1 Start writing the Dissertation	34 days	Mon 16/07/12	Thu 30/08/12
14.2 Submit the final product	1 day	Thu 30/08/12	Thu 30/08/12
14.3 Submit the Dissertation	1 day	Thu 30/08/12	Thu 30/08/12
15 End of Project	0 days	Thu 30/08/12	Thu 30/08/12



8.4.2. Updated



11 Monitoring	0 days	Fri 08/06/12	Fri 08/06/12
12 Controlling	41.06 days	Mon 28/05/12	Tue 24/07/12
12.1 Progression Status	41.06 days	Mon 28/05/12	Tue 24/07/12
12.1.1 Progress Report 1	1 hr	Fri 15/06/12	Fri 15/06/12
12.1.2 Writing Interim Report	15 days	Mon 28/05/12	Fri 15/06/12
12.1.3 Progress Report 2	1 hr	Fri 22/06/12	Fri 22/06/12
12.1.4 Progress Report 3	1 hr	Fri 29/06/12	Fri 29/06/12
12.1.5 In Progress Presentation	30 mins	Tue 24/07/12	Tue 24/07/12
12.1.6 Progress Report 4	1 hr	Fri 06/07/12	Fri 06/07/12
12.1.7 Progress Report 5	1 hr	Fri 13/07/12	Fri 13/07/12
13 Preparing the final project and starting the Dissertation	0 days	Mon 16/07/12	Mon 16/07/12
14 Closing	34 days	Mon 16/07/12	Thu 30/08/12
14.1 Start writing the Dissertation	34 days	Mon 16/07/12	Thu 30/08/12
14.2 Submit the final product	1 day	Thu 30/08/12	Thu 30/08/12
14.3 Submit the Dissertation	1 day	Thu 30/08/12	Thu 30/08/12
15 End of Project	0 days	Thu 30/08/12	Thu 30/08/12



8.5. Appendix E (Project Management Requirements)

8.5.1. Meetings' Minutes

1st Meeting's Minutes

28 Feb 2012
11:15 am – 11:45 am
Room 9.18
EC Stoner Building
University of Leeds

Present	Dr. Raymond Kwan - <i>Supervisor</i> Abdulaziz Albatli – <i>Student</i>
----------------	--

Number	Key Topics	Actions
1	Project The student has explained the area of interest in terms of the project. A project related to Tracsis Plc has been discussed and a meeting will be set with Darran Moss.	To meet with Darran Moss to have a clear picture of what's required.
2	Date of the Next Meeting 7 – Mar – 2012	The student should read about the company and have notes prepared in order to discuss with Darran Moss.

2nd Meeting's Minutes

7 Mar 2012
10 am – 11 am
Room 9.18
EC Stoner Building
University of Leeds

Present	Dr. Raymond Kwan - <i>Supervisor</i> Darran Moss - <i>Client</i> Abdulaziz Albatli – <i>Student</i>
----------------	---

Number	Key Topics	Actions
1	Current Process Reasons behind gathering data have been explained. Also, the procedures and processes undertaken to gather data have been explained. The client has also explained how the data on forms are converted to digital excel files by using Optical Character Recognition (OCR).	None
2	Initial Requirements The client has explained what he initially expects to see, some of which are: - The system should be web-based. - The system should be friendly to use. - The system should be adaptable and configurable to the types of surveys, only three are required. - The system should be able to take many files and store them in the DB and do analysis on them. - The system should have an authentication process for security (login procedure). - The graphical forms required: Graphs, Tables, and Charts. - For example, analysis could be: - o Average/Total number of passengers on a particular line/particular train/particular season. - Full Testing and Evaluation of the system should be undertaken before handing it to the client.	The student should prepare a Requirements Catalogue for the next meeting to be agreed by all parties.
3	Project Environment The timeline of the project period has been explained to the client.	None
4	Date of the Next Meeting 3 – Apr – 2012	The student should prepare a sketch of some suggested graphical forms and of the system.

3rd Meeting's Minutes

15 Mar 2012

1:00 pm – 1:40 pm

Room 9.18

EC Stoner Building

University of Leeds

Present	Dr. Raymond Kwan - <i>Supervisor</i> Abdulaziz Albatli - <i>Student</i>
----------------	--

Number	Key Topics	Actions
1	Project Management Discussed relevant documents: 1. WBS – Gaunt Chart. 2. Meetings Minutes. 3. Weekly Reports to monitor progress, starting from June 2012. 4. Time and Scope Constraint. 5. Risk Management Plan.	To all be produced by Aziz.
2	Literature Review Sources As a starting point, the student (Aziz) should read past project dissertations that are relevant to the undertaken project in order to have an insight of the suitable references and sources of information to build up the review.	To be undertaken by Aziz.
3	Meetings' Minutes The appropriate style of the form has been discussed as well as how and when to produce it.	To be undertaken by Aziz after every meeting.
4	Date of the Next Meeting	To be specified.

4th Meeting's Minutes

3 Apr 2012

9:30 am – 12 pm

Tracsis Plc

Derby, DE24 8ZS

Present	Dr. Raymond Kwan - <i>Supervisor</i> Darran Moss - <i>Client</i> Abdulaziz Albatli - <i>Student</i>
----------------	---

Number	Key Topics	Actions
1	Project Plan The proposed WBS (Work Breakdown Structure) and Gantt Chart have been discussed and approved by the Supervisor.	The student should update it, if necessary.
2	Updated Requirements Adding to the initial Requirements Catalogue: - The web system will be of three components, Admin, User, and Reporting system; - o The Admin should have control of the uploaded files and registering the users. - o The user should be able to generate the needed visual forms only from their relevant projects. - o Report Writing System (i.e. comment box) should be available for each generated visual form and for each uploaded project. - The system should support the principle of thought-navigation. For example, a pie chart may trigger more questions, so there should be more data available to answer possible questions, i.e. by having them on the same page, another page or on another visual form. - There should be a feature where one file could be re-uploaded and the older	The student should update the Requirements Catalogue

	one automatically deleted. - The Admin should be able to add details for each project; name, location, relevant clients, a date and a reference number. - The system should be expandable and non-restricted in terms of the DB. - There could be auto-generated visual forms once a project is uploaded.	
3	Sketches of the Web System's Design Initial designs of the web system have been discussed and comments were documented.	The student should consider the feedback while building the structure of the web system.
4	Next Meeting Mid of June 2012	- The student will have built the structure of the web system and the database. - By end of May, the student should contact the client and the supervisor to arrange a meeting.

5th Meeting's Minutes

12 Jun. 12
11:15 am – 11:45 am
Room 9.18
EC Stoner Building
University of Leeds

Present	Dr. Raymond Kwan – <i>Supervisor</i> Abdulaziz Albatli – <i>Student</i>
----------------	---

Number	Key Topics	Actions
1	Project Progress The updated WBS (Work Breakdown Structure) and have been discussed. The progress of building the web system and database have been discussed.	The student may carry on progressing according to the project plan.

	The design of the web system has been approved by the supervisor.	
2	Next Meeting 20 th June 2012	To brief the client with the progress of the project.

6th Meeting's Minutes

20 Jun 2012

1 pm – 1:50 pm

Tracsis Plc

Leeds Innovation Centre

Present	Dr. Raymond Kwan - <i>Supervisor</i> Darran Moss - <i>Client</i> Abdulaziz Albatli - <i>Student</i>
----------------	---

Number	Key Topics	Actions
1	In Progress Briefing The client has been briefed on what has been achieved on the web system. Feedback has been logged: - Web system design. - Functionality of the system - Some modifications in the database in terms of recursive link in the project table.	The student has to implement the feedback gained from this session.
2	Next Meeting (in Derby) 4 July 2012 to meet Anna (evaluation) and Charlie (design) 10 July 2012 9:30 am to meet Darran	- Evaluation on 4 July - Final meeting on 10 July.

7th Meeting's Minutes

4th July 2012

10:30 am – 12 pm

Tracsis Plc

Derby

Present	Anna Lucas – <i>Client</i> Charlie Simpson - <i>Client</i> Abdulaziz Albatli - <i>Student</i>
----------------	---

Number	Key Topics	Actions
1	In Progress Briefing The client has been briefed on what has been achieved on the web system in terms of the analysis of data. Feedback has been logged: - The exact type of analysis has been explained with regards to each type of the three files. - The exact visual forms have been discussed (Line chart, Pie Chart and Table). - A further requirement (enhancement) has been clarified in terms of comparing two seasons. - The layout of the web system will be changed once the website template has been received in time from Charlie.	The student has to implement the feedback gained from this session.
4	Next Meeting (in Leeds) 10 th July 2012 9:30 am to meet Darran	

8th Meeting's Minutes

10 July 2012

9:45 am – 11:45 am

Room 9.18

EC Stoner Building

University of Leeds

Present	Dr. Raymond Kwan - <i>Supervisor</i> Darran Moss - <i>Client</i> Abdulaziz Albatli - <i>Student</i>
----------------	---

Number	Key Topics	Actions
1	In Progress Briefing The client has been briefed on what has been achieved on the web system. Final Feedback has been logged: - Web system design. - Functionality of the system Further Enhancements has been agreed for both Project comparison and System Management and has been logged in the requirements catalogue.	The student has to implement the feedback gained from this session.
2	Evaluation Once the system has been fully implemented and deployed online, a number of people will experiment with the system and evaluate it.	- The student has to deploy the system online and send the url to the client to commence evaluation

8.5.2. Weekly Progress Reports

Supervisor Name: Dr. Raymond Kwan

Student ID: 200626611

Student Name: Abdulaziz Albatli

Week No: 1

Report No: 1

Date: 11 – 15/JUN/2012

Tasks undertaken & progress made against project plan	Action to be taken for next week
- Further background reading. - Finalizing Interim Report. - Going through similar already built web systems in terms of website template and database design. - Start designing and building the database. - Start building the web system in terms of web pages.	- Further background reading. - Finalizing the database. - Continue building the web pages.

Supervisor Name: Dr. Raymond Kwan

Student ID: 200626611

Student Name: Abdulaziz Albatli

Week No: 2

Report No: 2

Date: 18 – 22/JUN/2012

Tasks undertaken & progress made against project plan	Action to be taken for next week
- Further background reading. - Finalizing the database. - Continue building the web pages (master pages for admin and client, login system, etc).	- Further background reading. - Enhancing the database. - Continue building the web pages (account management, upload feature, data insertion, etc).

Supervisor Name: Dr. Raymond Kwan

Student ID: 200626611

Student Name: Abdulaziz Albatli

Week No: 3

Report No: 3

Date: 25 – 29/JUN/2012

Tasks undertaken & progress made against project plan	Action to be taken for next week
- Further background reading. - Enhancing the database. - Continue building the web pages (account management, upload feature, data insertion, and analysis part).	- Further background reading. - Enhancing the database. - Continue building the web pages (analysis part of the system).

Supervisor Name: Dr. Raymond Kwan

Student ID: 200626611

Student Name: Abdulaziz Albatli

Week No: 4

Report No: 4

Date: 2 – 6/JUL/2012

Tasks undertaken & progress made against project plan	Action to be taken for next week
- Further background reading. - Enhancing the database. - Continue building the web pages (analysis part of the system).	- Further background reading. - Enhancing the database. - Continue building the web pages (analysis part of the system). - Further Enhancement (Projects comp/arison).

Supervisor Name: Dr. Raymond Kwan

Student ID: 200626611

Student Name: Abdulaziz Albatli

Week No: 5

Report No: 5

Date: 9 – 13/JUL/2012

Tasks undertaken & progress made against project plan	Action to be taken for next week
- Further background reading. - Enhancing the database. - Continue building the web pages (analysis part of the system). - Further Enhancement (Projects comparison).	- Finalizing the build of the web system. - Deploying the system to a web host.

8.5.3. Project Risk Management

By reference to the Project Risk Assessment Probability & Consequence Coding Criteria provided below this table, I have completed the following table of Project Risks applied to “Visualisation of Rail Passenger Survey Data and its Delivery as a Web Service”.

Priority = A × B	A Probability of Risk Happening	B Consequences if Risk Happens	Description of Hazard	Source of Hazard	Action to be taken to Reduce Probability of an Accident Happening	Action to be taken to Reduce Consequences if an Accident does Happen	Action to be Taken in Event an Accident does Happen
9	3	3	Getting sick	Health Issues	Take care of health and safety	Consider back up plan in case health issues arises. Be under Health Insurance.	Spread the work over a number of days in order not to affect the progress. Minimizing the project if needed
5	5	1	Not attending the meetings	Careless	Giving advice and encouragement and having a contract to be signed by all the members.	Consider another plan	To be contacted by supervisor.

8	4	2	The scope plan and requirements might be beyond my skills and takes longer time than anticipated.	Over estimating	Try to design an adequate plan based on my skills and the time given as much as possible.	Considering another plan. Seek for advice from tutors. Finding other resources.	Minimize the scope plan as long as the basics of the project are met.
---	---	---	---	-----------------	---	--	---

Priority = A × B	A Probability of Risk Happening	B Consequences if Risk Happens	Description of Hazard	Source of Hazard	Action to be taken to Reduce Probability of an Accident Happening	Action to be taken to Reduce Consequences if an Accident does Happen	Action to be Taken in Event an Accident does Happen
2	2	1	Losing some of the work	Memory & devices break down or lost. Online backup crashes.	Trying to use efficient memory storage and programs. Saving the work before shutting down the Computer. Use multiple backup locations.	Regular back up. Multiple copies of the original work (Personal PC, Memory Stick, University Storage space, Dropbox)	Get the work from the updated back up or from the other copies made.
16	4	4	Client refuses to cooperate with data.	Client.	Initial agreement before starting the project.	Use test data as close as possible to the real data.	Contact supervisor for immediate action.
			Other risks to be defined later				

Project Risk Assessment Probability & Consequence Coding Criteria

Probability of Risk Happening Code	Probability of Risk Happening Qualitative	Probability of Risk Happening Quantitative
1	Nearly Certain	More than a 1 in 3 chance
2	Highly Probable	Between a 1 in 10 and a 1 in 3 chance
3	Moderately Probable	Between a 1 in 100 and a 1 in 10 chance
4	Possible but not very probable	Between a 1 in 1000 and a 1 in 100 chance
5	Unlikely but still not improbable	Between a 1 in 10,000 and a 1 in 1,000 chance

Consequences of Risk Happened Code	Consequences of Risk Happened
1	Overrun the Client's most pessimistic time and budget constraints and fail to achieve the Client's minimum expectations for the Product resulting in it being seriously deficient in a vital aspect of its performance.
2	Overrun the Client's time and budget, or overrun the Client's time or budget and fail to achieve the Customer's minimum expectations for the Product resulting in it being significantly deficient in an important aspect of its performance.
3	Overrun the Client's time or budget or fail to achieve the Client's expectations for the Product resulting in it being deficient in a significant aspect of its performance.
4	Jeopardises Time or Budget or the achievement the Client's complete expectations for the Product or seriously jeopardises the completion of a deliverable.
5	Compromises the Client's Time or Budget or the achievement of the Client's complete expectations or compromises the completion of a deliverable.

2012

8.6. Appendix F (User Guide)

Leeds University
Tracsis Plc

Abdulaziz Albatli

[USER GUIDE]
WWW.TRAC SISVISUAL.CO.CC

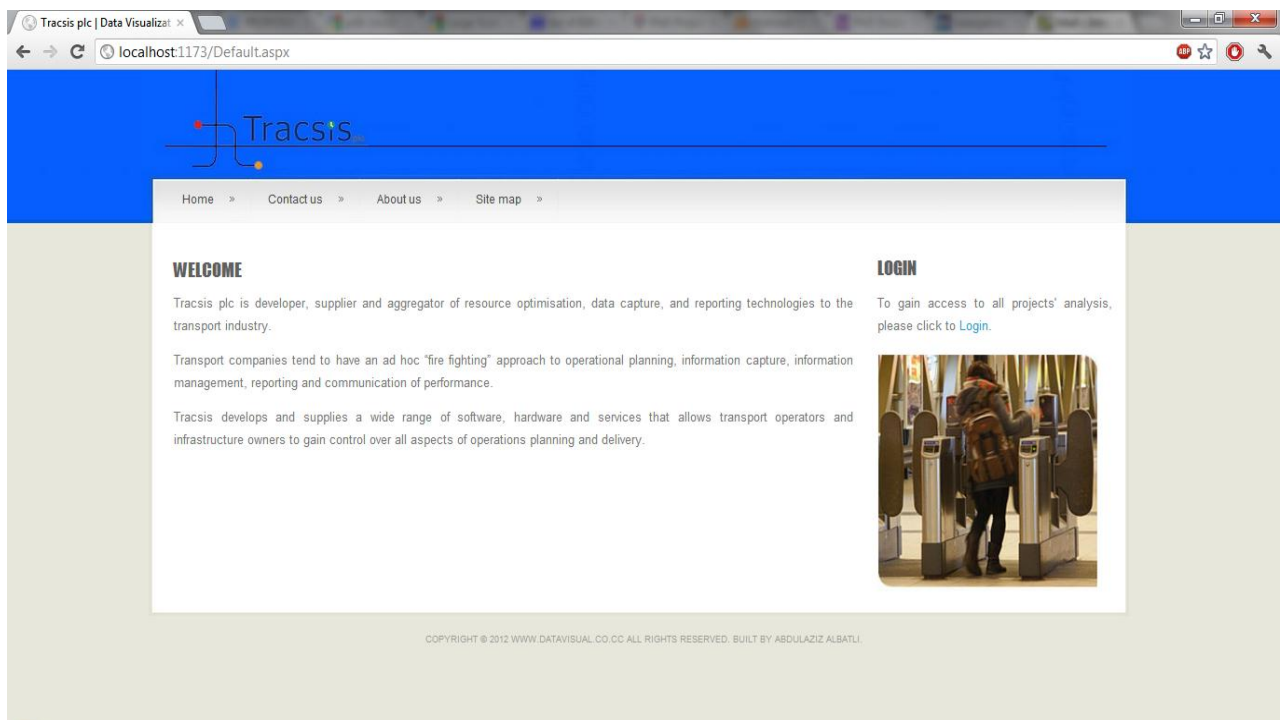
Content

Title	Page
About the system	2
General user	3
Admin user	5
Client user	14

About The System

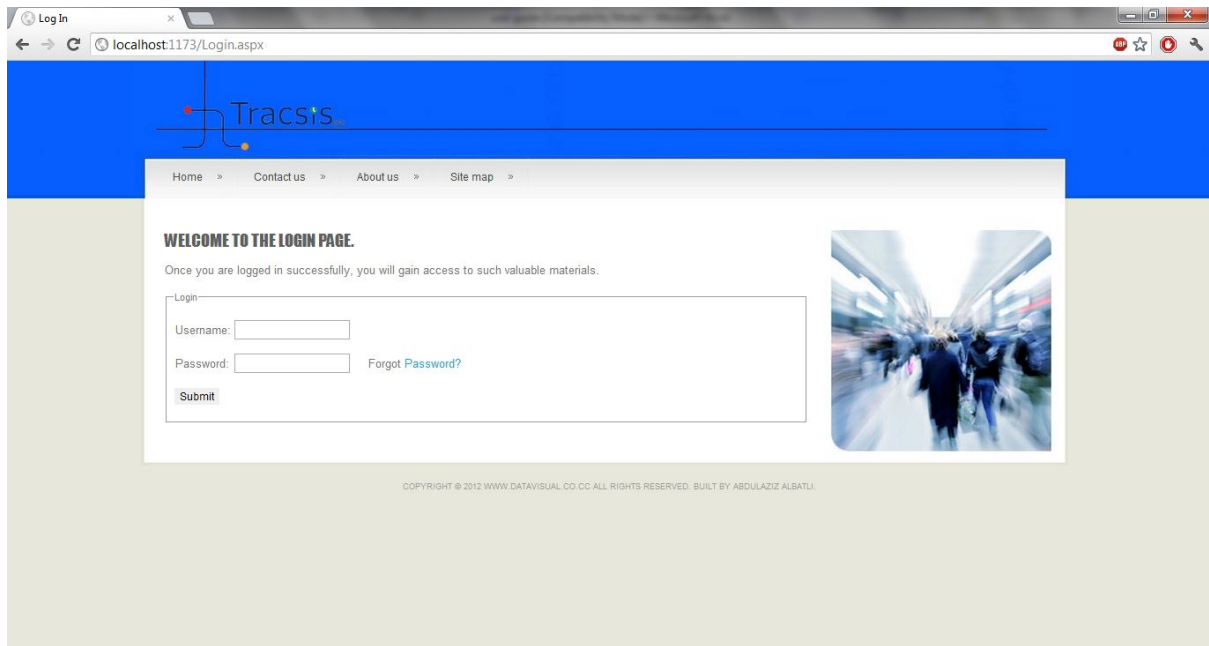
This web-based system provides visualization analysis of raw data, gathered by Tracsis Plc, fed into it. The raw data are generally about the number of passengers getting on/off trains and in/out train stations. Specifically, this system is a prototype of some aspects of data visualization, such as charts, graphs and tables. In addition, the main users of the system would be the clients of Tracsis Plc, i.e. rail companies and Department for Transport.

General User



Home Page

Home Link	It redirects the user to the Home Page
Contact Us Link	It redirects the user to the contact us page
About Us Link	It redirects the user to the about us page
Site Map Link	It redirects the user to the site map page
Login Link	It redirects the user to the login page



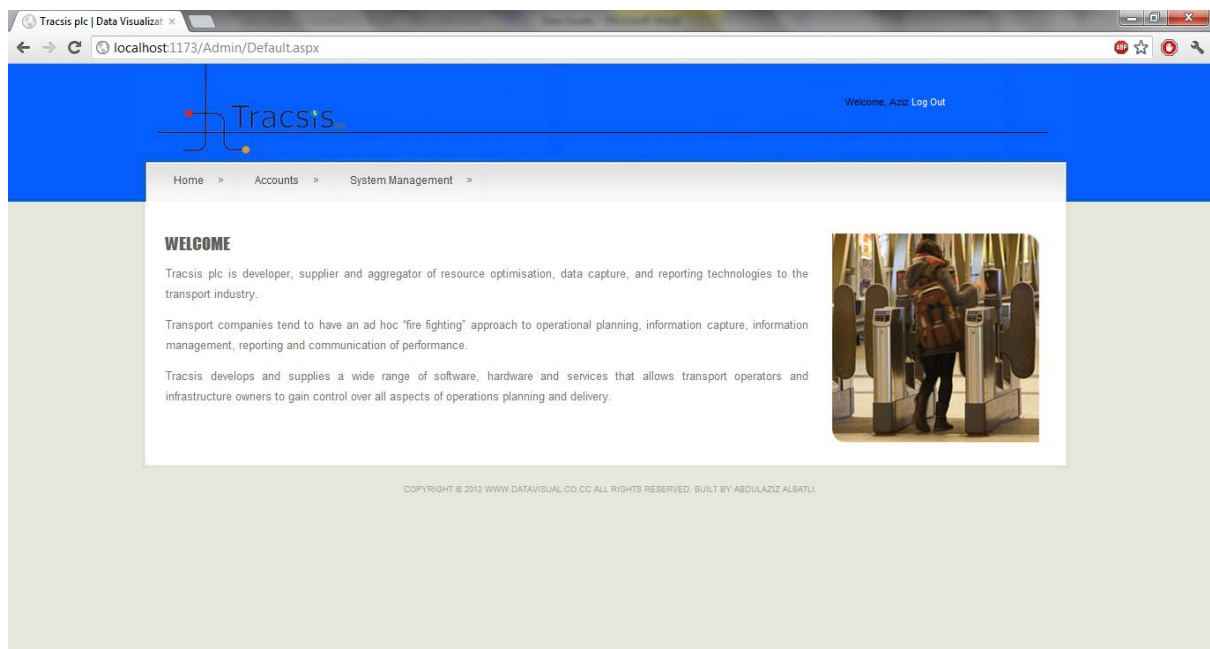
Login Page

Submit Button

Forgot Password Link

The user will fill in his log in details and then click on Submit to log in. It will redirect the user to the password retrieval page where he enters the username and then the login details will be sent to the registered email address.

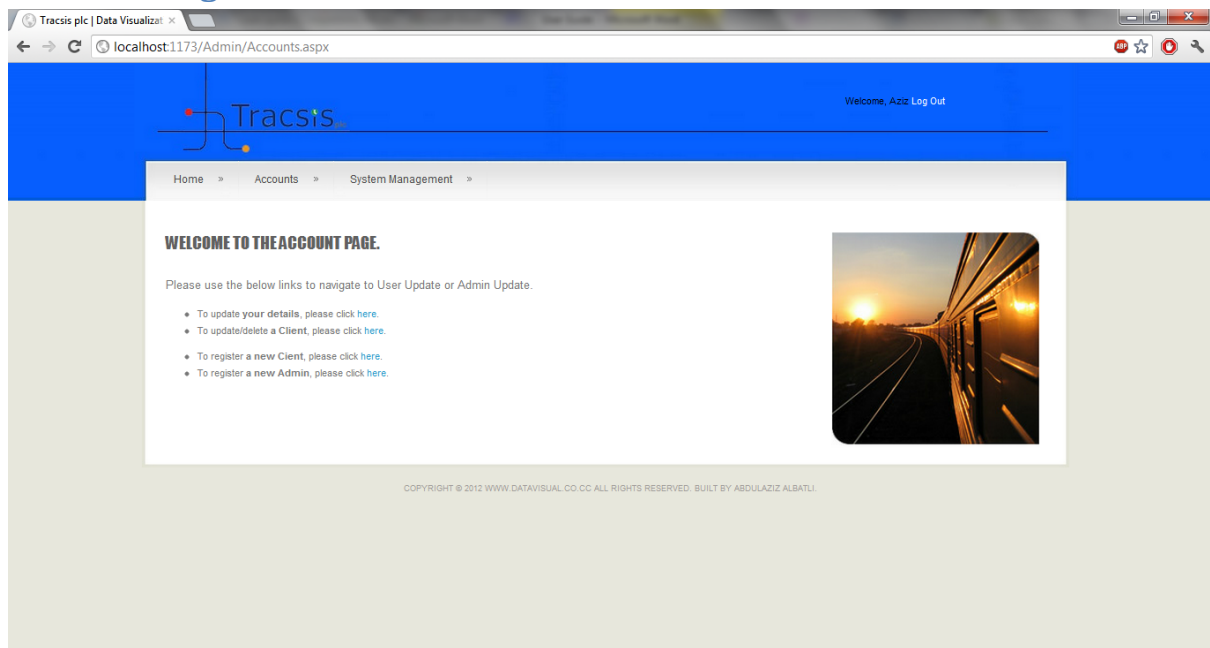
Admin User



Home Page

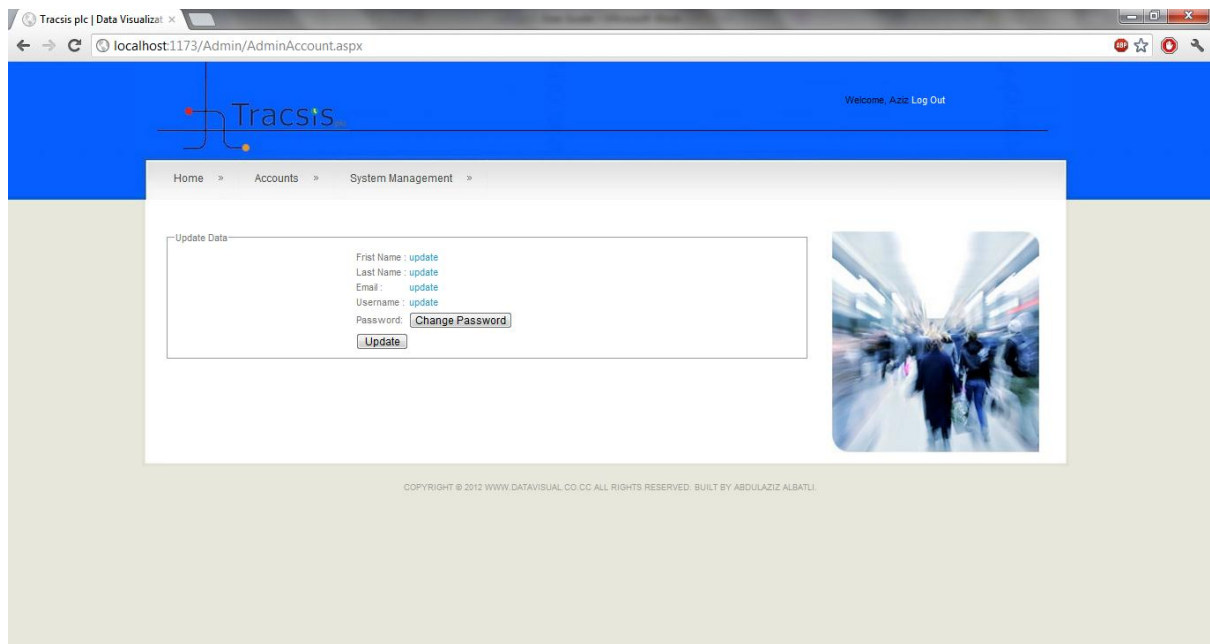
Home Button	It redirects the user to the Home Page
Accounts Button	It redirects the user to the Accounts page
System Management Button	It redirects the user to the System Management page
Logout Link	It will end the current session and logs out the user

Accounts Pages



Accounts Page

Update Details Link	It redirects the user to the Admin Update Details page
Update/Delete a Client Link	It redirects the user to the Client Update Details page
Register a New Client	It redirects the user to the Client Registration page
Register a New Admin	It redirects the user to the Admin Registration page



Admin Update Details Page

Update Links	Once clicked, data will be retrieved and shown in the relevant textbox
Update Button	It updates the database with the new data

Change Password
Button

It redirects the user to the Change Password page

Tracsis plc | Data Visualizat

localhost:1173/Admin/ClientAccount.aspx

Welcome Aziz Log Out

Home > Accounts > System Management >

CLIENTS ACCOUNTS PAGE

Please choose a client from the drop down menu.

Update/Delete Client

aziz Duncan

Select

Name:

Phone:

E-mail:

Username:

Password:

Update Send Details

Select a client to delete:

aziz Duncan

Delete Client

COPYRIGHT © 2012 WWW.DATAVISUAL.CO.CC ALL RIGHTS RESERVED. BUILT BY ABDULAZIZ ALBATLI

Client Update Details Page

Select Button

Once clicked, data for the selected client will be retrieved and shown in the relevant textboxes

Update Button

It updates the database with the new data

Send Details Button

Once clicked, an email will be sent with the login details. YOU have to choose (click on) the client first.

Delete Client

The selected client will be deleted from the database

Tracsis plc | Data Visualizat

localhost:1173/Admin/ClientRegistration.aspx

Welcome Aziz Log Out

Home > Accounts > System Management >

CLIENT REGISTRATION

Please use the below form to register a new Client.

Registration

Client Name:

Client E-mail:

Client Phone:

Client Username:

Client Password:

Register

COPYRIGHT © 2012 WWW.DATAVISUAL.CO.CC ALL RIGHTS RESERVED. BUILT BY ABDULAZIZ ALBATLI

Client Registration Page

Register Button

Once clicked, a new client will be registered in the system

Tracsis plc | Data Visualizat

localhost:1173/Admin/AdminRegistration.aspx

Welcome, Aziz Log Out

Home » Accounts » System Management »

ADMIN REGISTRATION

Please use the below form to register a new Admin.

Registration

Admin Firstname:

Admin Lastname:

Admin E-mail:

Admin Username:

Admin Password:

COPYRIGHT © 2012 WWW.DATAVISUAL.CO.CC ALL RIGHTS RESERVED. BUILT BY ABDULAZIZ ALBATLI

Admin Registration Page

Register Button

Once clicked, a new admin will be registered in the system

System Management Pages

Tracsis plc | Data Visualizat

localhost:1173/Admin/SysMgt.aspx

Welcome, Aziz Log Out

Home » Accounts » System Management »

SYSTEM MANAGEMENT

- To register a new Project, please click [here](#).
- To register a new Job, please click [here](#).

YOUR PROJECT LIST

Project Name	Client Name	Project Location	Project Reference	Project Date	Analysis	Edit	Delete
Leeds2012	Duncan	Leeds	ole8ui	26-06-2012	Analysis		
london232	aziz	London	iju8i	26-06-2012	Analysis		

YOUR JOB LIST

Job Name	Client Name	Job Location	Job Season Type	Add Project	Analysis	Edit	Delete
Spring 10-11	aziz	Reading	Spring	Add Project	Analysis		

COPYRIGHT © 2012 WWW.DATAVISUAL.CO.CC ALL RIGHTS RESERVED. BUILT BY ABDULAZIZ ALBATLI

localhost:1173/Admin/SysMgt.aspx

System Management Page

This page is where projects and jobs can be created. It shows previously created projects/jobs in two tables.

Register a New Project Link

It redirects the user to the Project Registration page

Register a New Job Link

It redirects the user to the Job Registration page

Analysis Button

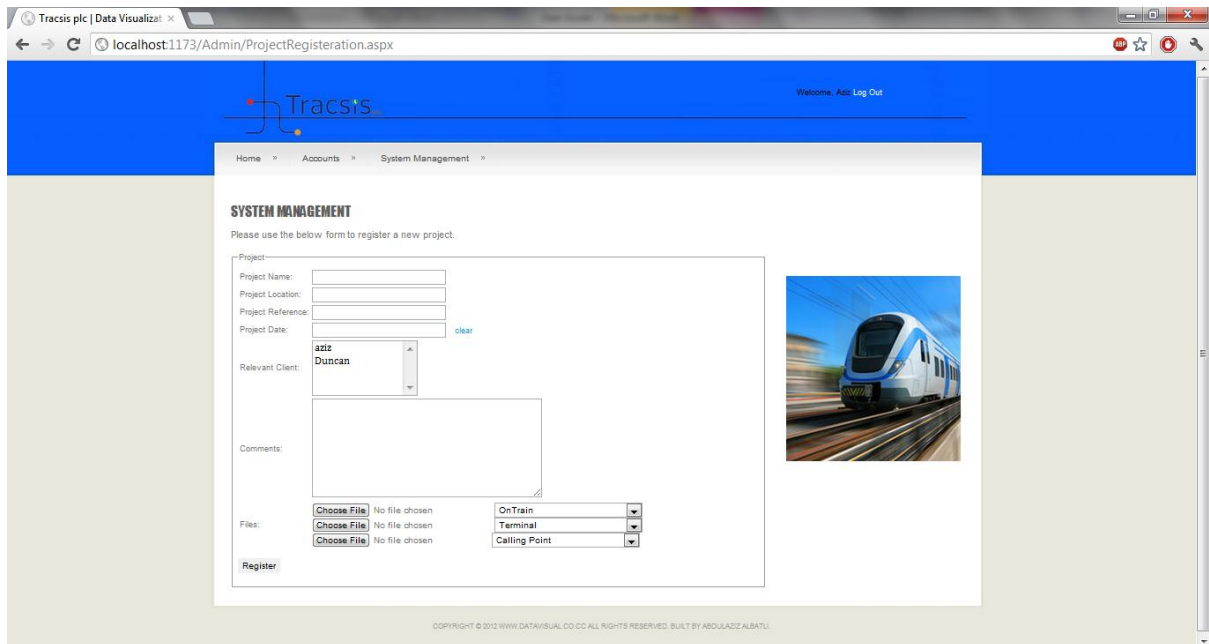
It redirects the user to the Visual Form Generation page

Edit Button

It redirects the user to the Project/Job Editing page

Delete Button

Once clicked, the relevant project/job will be deleted



Project Registration Page

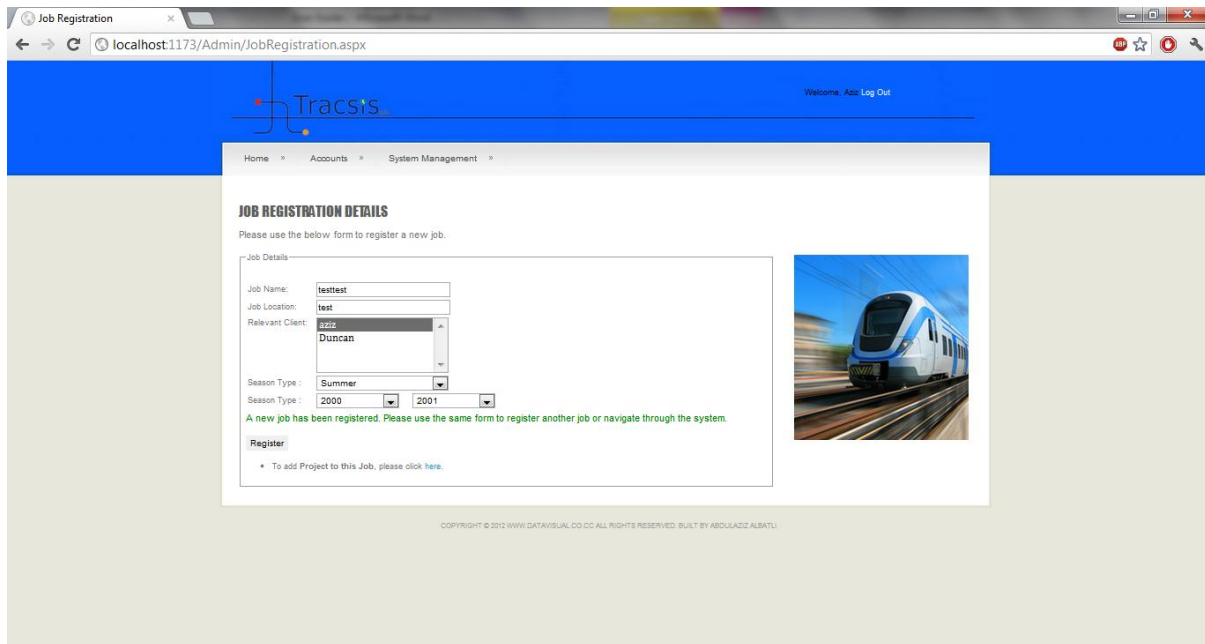
In this page, a new project can be registered. This form let the user to select the related client. Also, it allows the user to upload up to three files for one project.

Choose File Button

Allows the user to navigate through his computer to find/upload a file

Register Button

Once clicked, a new project will be registered in the system and the file(s) will be stored in the database.



Job Registration Page

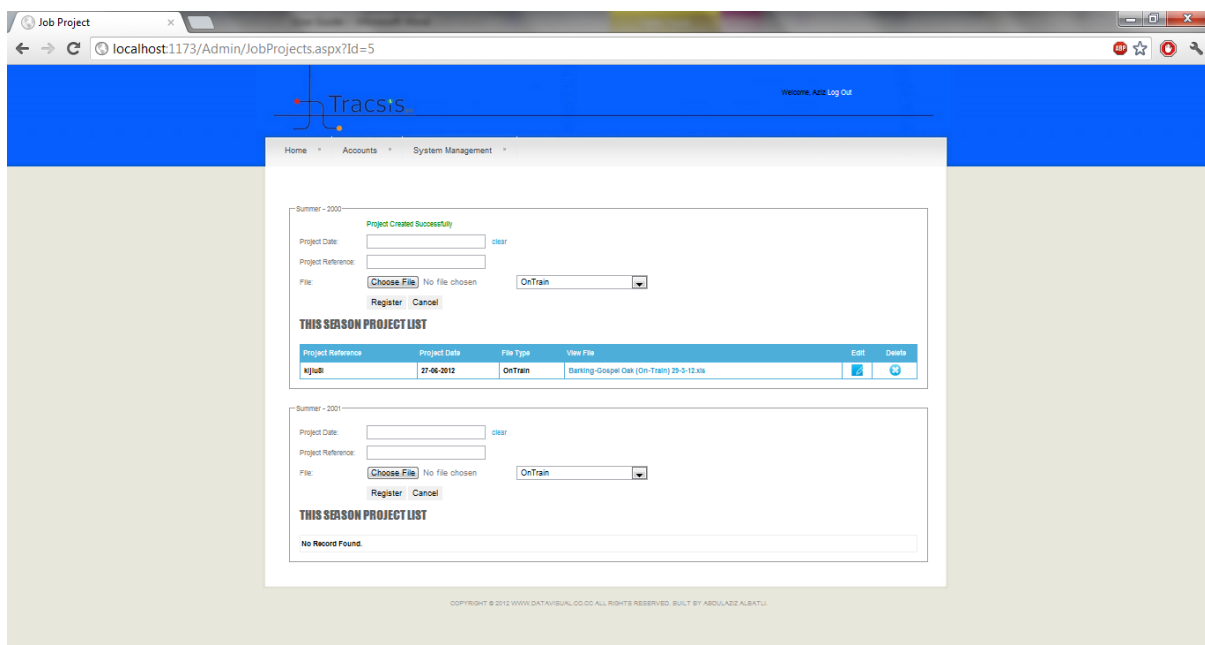
In this page, a new job can be registered. This form let the user to select the related client. Also, it allows the user to specify the season and the relevant years in order to for their projects to be compared.

Register Button

Once clicked, a new project will be registered in the system

Add Project Link

It redirects the user to a page where he can create the relevant projects for each season.



Job Registration Page – Project Creation

In this page, the user is allowed to create a number of projects per season. Also, s/he can modify/update the created projects using the edit button

Register Button

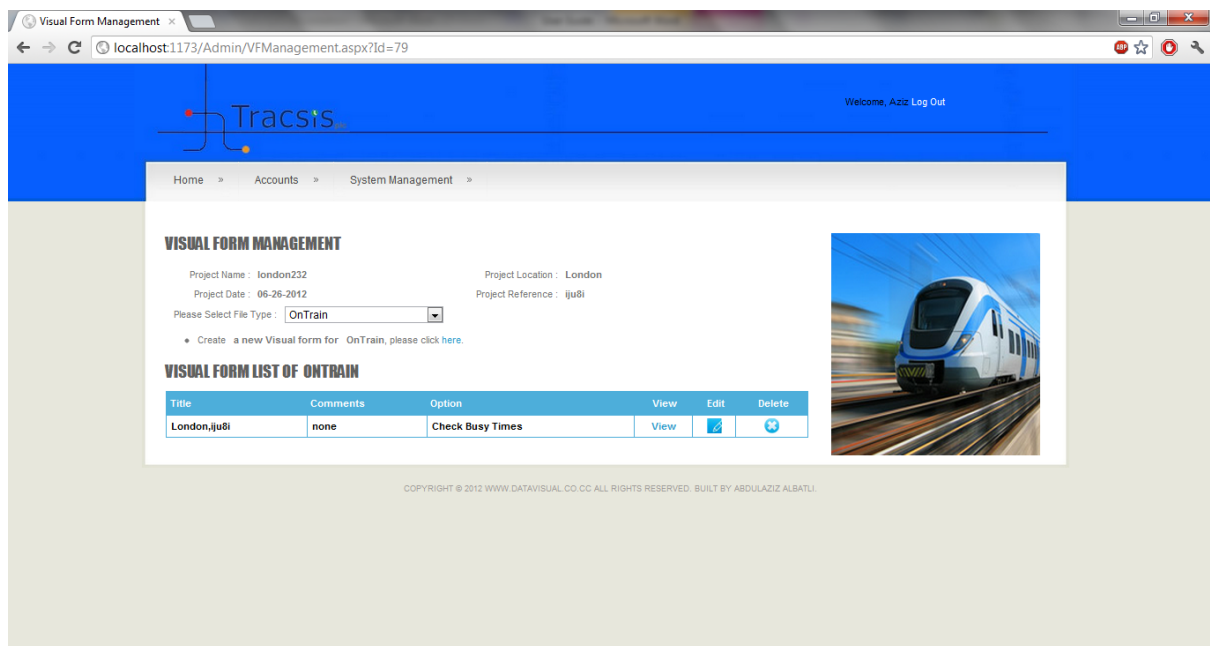
Once clicked, a new project will be registered in the system

View File Link

It opens the uploaded file in Microsoft Excel.

Edit Button

It allows the user to edit the project details and replace the uploaded file.



Visual Form Generation Page

In this page, previously generated Visual Forms will be shown in a table. Also, it lets the user chooses which file type to do analysis on.

File Type Dropdown Menu

Specifies which file, in case a project has different types of files, to be analyzed

Visual Form Creation Link

It redirects the user to the Visual Form Configuration Page.

View Button

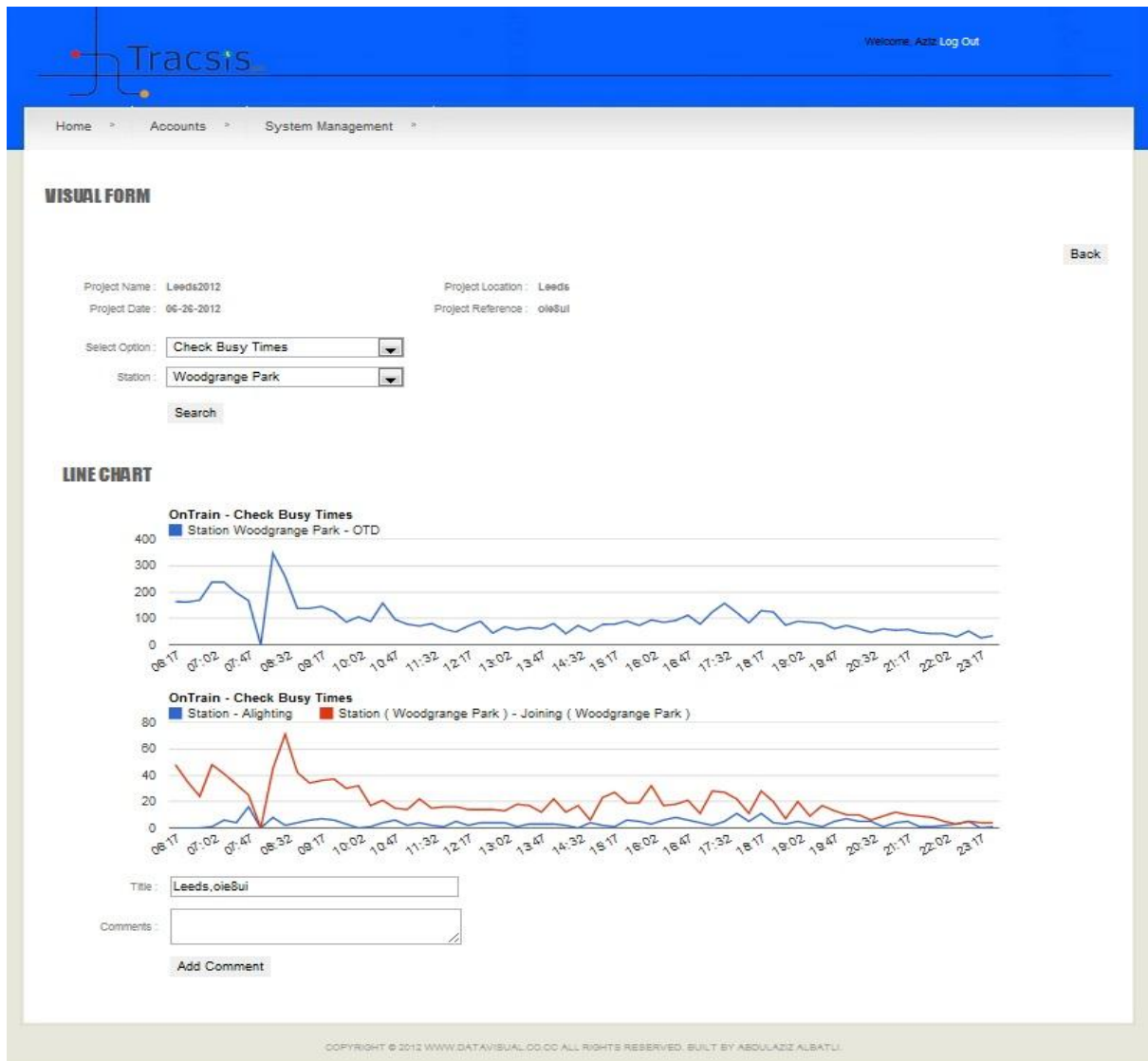
Once clicked, the relevant Visual Form(s) will be displayed in a printable format.

Edit Button

Once clicked, the relevant Visual Form(s) will be displayed and their comments can be edited.

Delete Button

Once clicked, the relevant Visual Form(s) will be deleted.



Visual Form Generation Page

In this page, there are two analysis options for every type which can be chosen from the Select Option dropdown menu. Once the Research Button has been clicked, the relevant Visual Forms will be generated. The user can amend the title and comment and can store them in the database by clicking Add Comment.

Select Option Dropdown Menu Specifies which analysis option to be commenced.

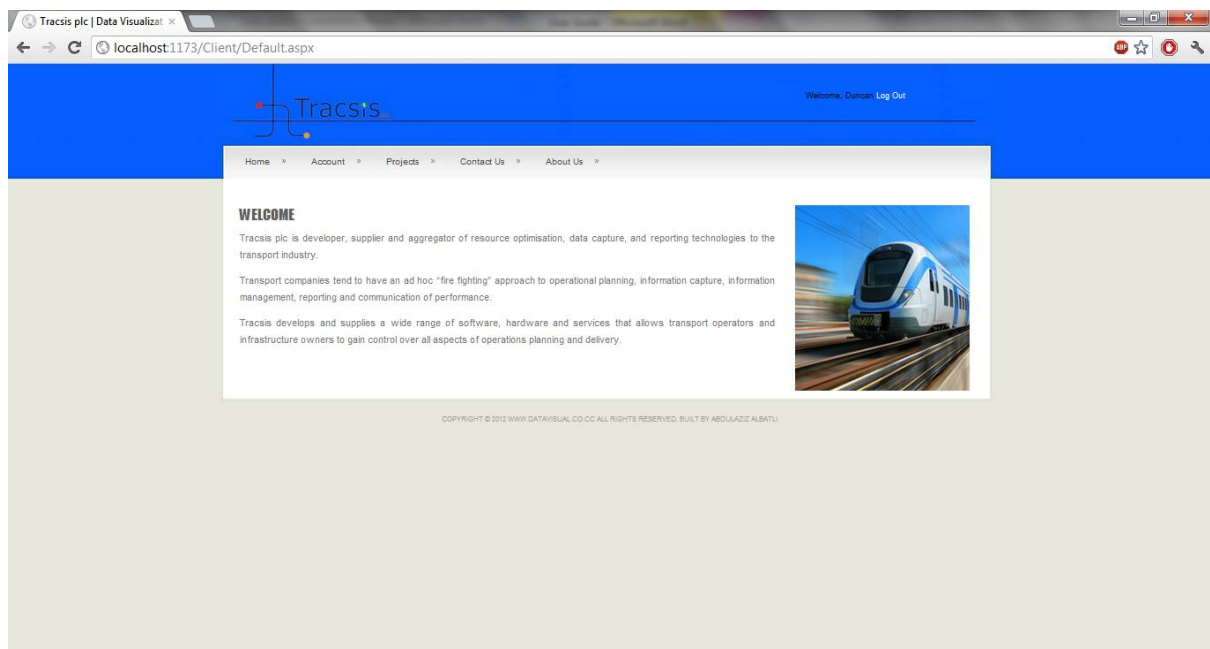
Search Button Generates the Visual Forms

Back Button It redirects the user to the previous page.

Job Page – Project Comparison

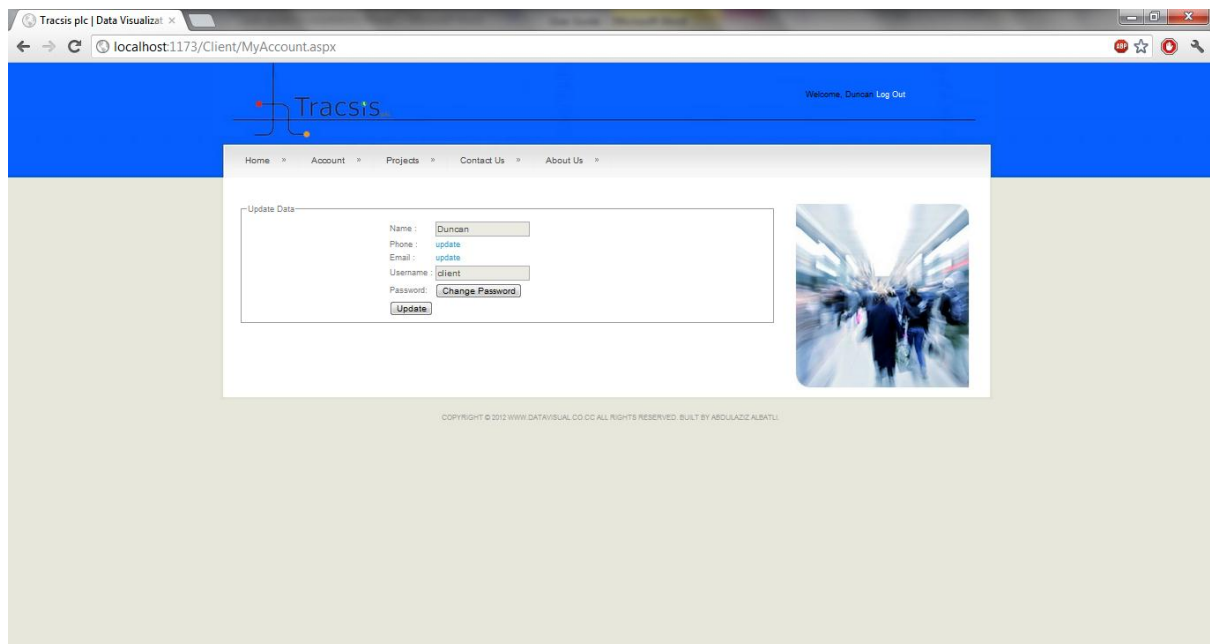
In this page, the user can specify two projects, from the same type, to be compared. Then, the same configuration analysis get displayed.

Client User



Home Page

Home Link	It redirects the user to the Home Page
Contact Us Link	It redirects the user to the contact us page
About Us Link	It redirects the user to the about us page
Account Link	It redirects the user to the account page
Projects Link	It redirects the user to the projects page

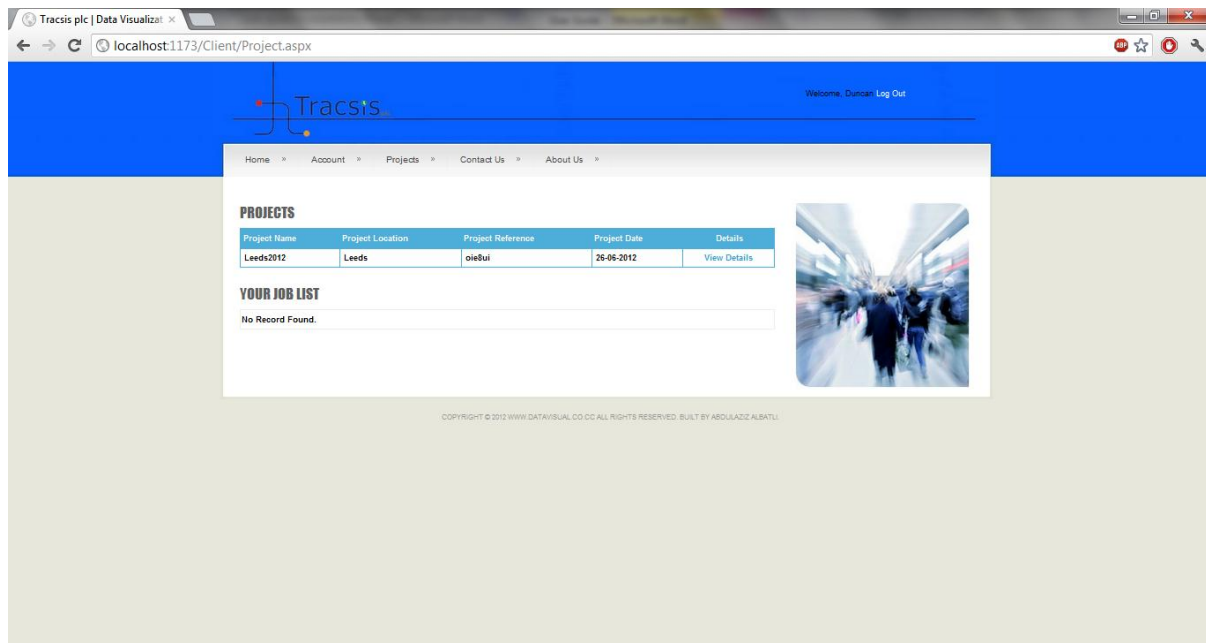


Client Update Details Page

Update Links	Once clicked, data will be retrieved and shown in the relevant textbox
Update Button	It updates the database with the new data

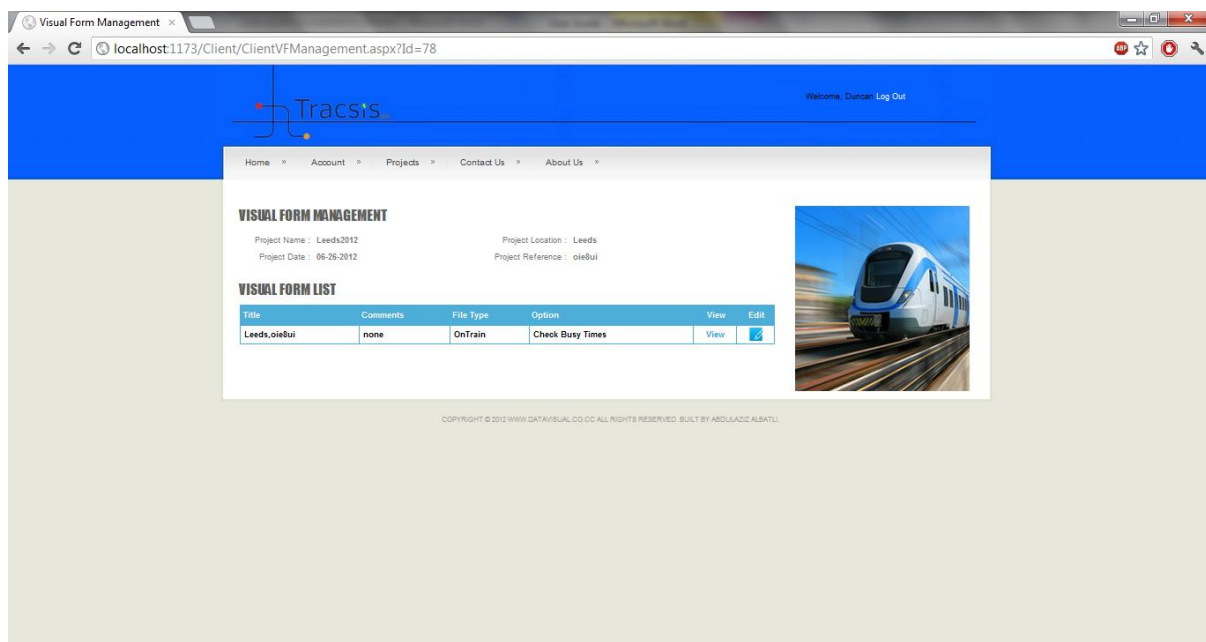
Change Password
Button

It redirects the user to the Change Password page



Project Page

In this page, the client can view his commissioned projects and jobs that have visual forms generated by admin. Once they click View Details, they can see the relevant visual formats for that project/job and may amend the Comment of everyone.



Client Update Details Page

Here, a list of visual forms already created is displayed. the view link is to show the visual form(s) in a printable format. Also, the Edit Button allows, once clicked, allows the user to see the visual form and edit its title and the comment

8.7. Appendix G (Manual Testing)

Object of Test	Expected Action	Pass or Fail	Comment
General User			
Home Button	It should redirect the user to the home page.	Pass	
Contact Us Button	It should redirect the user to the contact page.	Pass	
About Us Button	It should redirect the user to the about us page.	Pass	
Site Map Button	It should redirect the user to the site map page.	Pass	
Login Page			
Login Link	It should redirect the user to the login page.	Pass	
Admin User			
Home Button	It should redirect the user to the home page.	Pass	
Accounts Button	It should redirect the user to the accounts page.	Pass	
System Management Button	It should redirect the user to the System Management page.	Pass	
Accounts Page			
Update your details Link	It should redirect the user to the admin account page.	Pass	
Update/delete a Client Link	It should redirect the user to the client account page.	Pass	
Register a new Client Link	It should redirect the user to the admin registration page.	Pass	
Register a new Client Link	It should redirect the user to the client registration page.	Pass	
Admin Account Page			
Update Button	It should update the database with the new details	Pass	
Cancel Button	It should abort the update process	Pass	
Change Password Button	It should redirect the user to the Change Password page.	Pass	

Client Account Page		
Select Button	It should retrieve the selected client's details.	Pass
Update Button	It should update the database with the new details	Pass
Send Email	It should enable an email to be sent to the selected client with the login details.	Pass
Delete Button	It should delete the selected client from the system	Pass
Admin and Clint Registration Pages		
Register	It should validate the input and if passed it should insert the new user details to the system	Pass
System Management Page		
Register Project Link	It should redirect the user to the project registration page.	Pass
Register Job Link	It should redirect the user to the job registration page.	Pass
Analysis Button	It should redirect the user to the Visual Form Generation page	Pass
Edit Button	It should redirect the user to the Project/Job Editing page	Pass
Delete Button	Once clicked, the relevant project/job should be deleted	Pass
Job/Project Registration Pages		
Choose File Button	Allows the user to navigate through his computer to find/upload a file	Pass
Add Project Link	It should redirect the user to a page where he can create the relevant projects for each season.	Pass
Register Button	Once clicked, a new project should be registered in the system and the file(s) should be stored in the database.	Pass
Visual Form Generation Page		
File Type Dropdown Menu	Specifies which file, in case a project has different types of files, to be analyzed	Pass

Visual Form Creation Link	It redirects the user to the Visual Form Configuration Page.	Pass
View Button	Once clicked, the relevant Visual Form(s) will be displayed in a printable format.	Pass
Edit Button	Once clicked, the relevant Visual Form(s) will be displayed and their comments can be edited.	Pass
Delete Button	Once clicked, the relevant Visual Form(s) will be deleted.	Pass
Select Option Dropdown Menu	Specifies which analysis option to be commenced.	Pass
Search Button	Generates the Visual Forms	Pass
Back Button	It redirects the user to the previous page.	Pass

Client User

Home Link	It redirects the user to the Home Page	Pass
Contact Us Link	It redirects the user to the contact us page	Pass
About Us Link	It redirects the user to the about us page	Pass
Account Link	It redirects the user to the account page	Pass
Projects Link	It redirects the user to the projects page	Pass
Update Links	Client Update Details Page Once clicked, data will be retrieved and shown in the relevant textbox	Pass
Update Button	It updates the database with the new data	Pass
Change Password Button	It redirects the user to the Change Password page	Pass
View Link	Projects Page It redirect the user the visual form page related to the select project	Pass
Edit Button	Once clicked, the relevant Visual Form(s) will be displayed and their	Pass

Delete Button

comments can be
edited.

Once clicked, the
relevant Visual
Form(s) will be
deleted.

Pass

8.8. Appendix H (Survey Forms)

System Evaluation 1

The goal of this survey is to determine how satisfying the system is and its level of dependability. The results will be used in my MSc Dissertation as part of the evolution stage.

Please answer the below question according to your level of agreement (tick):

Question	Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
1. The site's design is aesthetically appealing.	X				
2. The information is sufficient for the intended audience.	X				
3. The organization of the site is logical and clear.			X		
4. The site is friendly and easy to use (i.e. navigation).	X				
5. The site is free from errors and bugs.				X	
6. The pages load reasonably quickly.	X				
7. I am happy with the overall web system.			X		

Please answer the below questions;

Please share any comments relevant to the above questions.

On a number of occasions the system showed up an error screen that meant I wasn't able to continue using it.

I would like to test the system out with clients to decide whether it is truly useful to them as I'm undecided as to whether there is too much presumption about how much they will already need to know and how much guidance we will need to give them initially. Part of my concern stems from the fact that I don't think the navigation and organisation of the site is as logical as it could be.

Did you expect something to be in the web system but was not implemented?

No

Turn Over

Have you found something that you did NOT like in the web system?

Other than the errors mentioned above, no

Have you found something that you strongly liked in the web system?

The variation in analysis available. That we can have complete control as administrators.

In your opinion, how dependable is the system?

Not very! I found I was getting error messages after using the system for about 5 minutes.

Please share any suggestions for future improvements of the web system.

The error messages need to go! Also, I trial with clients might be useful to assess how user friendly it is from their perspective.

System Evaluation 2

The goal of this survey is to determine how satisfying the system is and its level of dependability. The results will be used in my MSc Dissertation as part of the evolution stage.

Please answer the below question according to your level of agreement (tick):

Question	Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
1. The site's design is aesthetically appealing.	Yes				
2. The information is sufficient for the intended audience.		Yes			
3. The organization of the site is logical and clear.		Yes			
4. The site is friendly and easy to use (i.e. navigation).			Yes		
5. The site is free from errors and bugs.			Yes		
6. The pages load reasonably quickly.	Yes				
7. I am happy with the overall web system.		Yes			

Please answer the below questions;

Please share any comments relevant to the above questions.

The system presents the data in a very clear and interactive way and is ideal for our clients. The system also responds quickly, and is generally easy to use, however some basic training is required, although this is to be expected.

Did you expect something to be in the web system but was not implemented?

No.

Turn Over

Have you found something that you did NOT like in the web system?

Not that I can think of.

Have you found something that you strongly liked in the web system?

The presentation of the data and the ability to hover over the graphs and see the breakdown.

In your opinion, how dependable is the system?

Very early days to give an opinion but nothing has caused concern yet.

Please share any suggestions for future improvements of the web system.

The ability to handle more survey types.

System Evaluation 3

The goal of this survey is to determine how satisfying the system is and its level of dependability. The results will be used in my MSc Dissertation as part of the evolution stage.

Please answer the below question according to your level of agreement (tick):

Question	Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
1. The site's design is aesthetically appealing.	X				
2. The information is sufficient for the intended audience.	X				
3. The organization of the site is logical and clear.	X				
4. The site is friendly and easy to use (i.e. navigation).			X		
5. The site is free from errors and bugs.	X				
6. The pages load reasonably quickly.	X				
7. I am happy with the overall web system.	x				

Please answer the below questions;

Please share any comments relevant to the above questions.

Some of the design was counter-intuitive and not clearly explained in the documentation. For instance the use of "Add Comment" to publish Visual Forms to be viewed by Clients was not easy to figure out.

Did you expect something to be in the web system but was not implemented?

No

Have you found something that you did NOT like in the web system?

Turn Over

The arrangement of the documentation to focus on what you see on each page of the website, rather than walking step by step through how to gain a desired result.

Have you found something that you strongly liked in the web system?

The ability to link projects under one Job, and then view the data from two projects side by side.

In your opinion, how dependable is the system?

Seems solid so far, but has not seen heavy use yet.

Please share any suggestions for future improvements of the web system.

Ability to view larger charts, so that more data can be shown without details becoming lost.